Final Project

Bacherlor's degree in Industrial Technology

Thermal study of a Raspberry Pi for the construction of a small-sized supercomputer

MEMORY

Autor: Marc Nogué Pujiula
Director: Vicente César de Medina Iglesias
Co-director: Elisabet Mas de les Valls Ortiz
Date: June 2016

ETSEIB

Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona

UPC

1 Abstract

This project is part of a series of projects whose final purpose is to design a very small computing cluster coupling up to 100 units of Raspberry Pi 2, equivalent to a system with 400 CPUs. The main advantage of a cluster like this it is its reduced size and a substantial cost-cutting on the investment.

As one of the two first projects of these series, the present one aims to thermally characterize the Raspberry Pi 2 and propose a feasible cooling technology. The most critical issue of such a cluster is the cooling design in order to ensure a proper CPU temperature. Indeed, if the temperature exceeds a threshold of 85ºC, the device can crash [1]. However, such temperature is too high to guarantee an optimal performance of the Raspberry. To this aim, our goal is to maintain the device temperature below 60ºC under extreme atmosphere conditions, i.e. at an air temperature of 40 ºC.

The first task within present study has been to experimentally detect and quantify hot spots in the Raspberry Pi 2, when working at full capabilities. This information is required to decide which points need a special cooling technology. In order to reduce the experimental costs, the viability of using a NTC thermistor and a microcontroller has been studied and compared with the classical type-K thermocouples connected with the data logger. To this aim, a preliminary experiment has been designed to calibrate a NTC thermistor

The second task has been to calculate the thermal power generated in the different components of the device. To this purpose, the data reported in the experiment for the detection of hot spots has been used. Also, the thermal convection coefficients required to calculate the thermal power generated have been estimated using available correlations in the literature.

The third and last task of the project has been to determine experimentally if the available heat transfer coefficients in the literature are valid in the presence of fins designed for small devices as the Raspberry Pi 2. This task includes two different experiments: one for natural convection and another one for forced convection using a small wind tunnel.

Some Arduino and Raspberry PI knowledge is required to carry out the three tasks. In the document, the basic information has been provided so the reader can manage to keep on the reading. Moreover the sketches are completely commented and several tutorials are attached to help the reader understand the content of this document easily.

With the successful accomplishment of the three tasks described above, the basic information required in order to proceed with the design and optimization of the very small cluster is available and subsequent studies can be performed.

2 Table of Contents

3 List of figures and tables

4 Preface

Mankind is now experiencing its fifth and most intense technological revolution, and we are transitioning into the Hybrid Age. Most people believe we are still living in the Information Age, but in fact we have already reached an inflection point, a brewing storm that will once again drastically change individual life and society. The so-called "Information Age" means that in real time we have access to what others share in platforms like Twitter, Facebook, Instagram, Snapchat, etc. It is a time when digital developments on the web or smartphones are improved day by day, and we must understand that we are witnessing an era in the history quite hard to overlook.

In order to achieve this constant development it is required to have the ability or, sometimes the luck, to couple the right software and the right hardware. And although the work in this field is surprisingly fast and amazing there is yet many technologies and devices to come.

This project makes the contribution towards catching this unstoppable train called progress.

All the online platforms such as Facebook, Instagram, Snapchat must have their own servers physically located in a place. The places were all this computers stay are called clusters. A cluster is nothing else but a set of computers connected to each other in order to work together. That, can be understood as a single system [2]. Each computer is set to perform the same task so, in broad terms, can be considered as a very powerful and efficient calculator.

Computer clusters early days are quite difficult to predict due to its poor accuracy of its definition at the beginning. They were not invented by any specific vendor but by customers who more power and storage needed space in their computers (see Figure 1). Greg Pfister estimates that this happened in the 1960s[3]. The more formal basis of the cluster computing as nowadays is understood, do parallel work of any sort, was invented in 1967 by Gene Amdahl of IBM. He published what is known as Amdahl's Law and it's a seminal paper on parallel processing. But it was not until 1984 when clustering really take off with the VAXcluster (see Figure 2) developed by Digital Equipment Corporation [4].

Figure 1. Home-built computer cluster

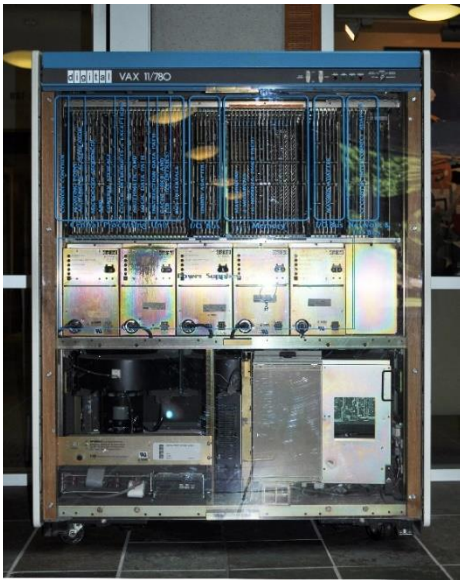

Figure 2. VAX 11/780. One of the first computer clusters in 1980s.

Computer clusters can be configured for a very wide range of works, starting from general purposes, such as web-service support, to more specific ones, i.e. big computation problems in the scientific field (see Figure 3). Depending on the work they are being designed for, they will require of different capacities. Bigger capacity of process data means more cost and more space.

Figure 3. Facebook servers in 2012.

4.1 Motivation

Nowadays huge stations are being used in order to process massive amounts of data. This stations need to be located in big infrastructures and also consume a lot of power. Unexpectedly these places aren't getting smaller as anyone might think after arguing that the technology tries to simplify human lives. In fact they are getting bigger, it has been estimated that nowadays Google has more than 2.000.000 servers distributed around the world [5]. Although it is true that the information that flows through the internet everyday increases and this means bigger storage space and more servers. But somehow, there has to be a way to keep up with this constant increasing amount of information whilst maintaining the dimensions of the clusters where it is processed.

The need to find a solution to this made this project suitable. Using a Raspberry Pi 2 as a processor of the data, one can simulate a cluster connecting several of this devices. The main difference between this microcomputer and the actual servers is that it is cheap and small. Nevertheless, the heat generation and a proper cooling system must be taken into account.

4.2 Scope

This project is a contribution to a bigger one whose goal is to create a cluster able to replace the actual workstations. The scope of this project is specifically the calculation of the heat generation and the convective heat transfer. It will also be determined those spots on the Raspberry Pi 2 that generate more heat. This is meant to be analysed experimentally at a laboratory.

The results obtained from this document are intended to be used in a further project in order to design a proper cooling system and assemble the devices accordingly to the best geometry possible ensuring a safe thermal configuration.

5 Introduction

5.1 Workflow

This project is only the first part of a bigger one. As mentioned before, the aim of this document is to detect the main heat generation spots and to estimate the thermal convection coefficient of the air when the device is running. The major steeps carried out to complete the project, ordered chronologically are:

1. Understand the Arduino IDE environment and its main features.
2. Design a calibration experiment for the NTC thermistor.
3. Validate the calibration of the NTC.
4. Design an experiment to spot the heat points and measure the generated thermal power on the Raspberry Pi.
5. Experimental campaign to spot the heat points and data analysis (thermal generation).
6. Design an experiment to calculate the thermal convection coefficient with and without fins.
7. Experimental campaign to calculate the thermal convection coefficient and data analysis.

5.2 Structure of the project

The workflow specified in the previous section is developed in sections 7, 8 and 9. At the same time this is divided into smaller subsections all of them addressed so can be found easily.

Each of the small sections pretend to explain, as understandable as possible, every step executed along this project. Basically the document can be split in two sections, the theory part and the experimental one. The theory part, developed in section 6, is needed in order to comprehend the experiments and calculations done in the experimental part. The document is intended to be easy to read with no excessive amount of theory.

6 Previous concepts of thermodynamics

The aim of this section is acquaint the reader with the thermodynamics concepts used in this document. It is a brief introduction into the convection and radiation mechanisms. All the correlations and examples used in this section are cited in the following chapters.

6.1 Thermal convection coefficient

One of the goals of this project is to determine the experimental thermal convection coefficient of the air in forced convection whilst using the Raspberry Pi. In order to do that some theory concepts must be considered. The following pages are meant to briefly explain those main concepts about convection.

Convection is the transfer of internal energy into or out of an object by the physical movement of a surrounding fluid that transfers the internal energy along with its mass. This energy transfer is due to a gradient on the temperatures of both object and fluid/gas.

The heat transfer convection is composed by two simultaneous mechanisms: The diffusion or conduction in a molecular scale and the global movement or macroscopic.

If the temperature of the object is higher than the fluid one, then its molecules have a higher energetic level than the one of the fluid as well. In this circumstance this energetic molecules give by simple contact or elastic collision part of their energy to the molecules of the fluid in contact with them. This increases the local temperature of the fluid, being higher than the one of the environment. Another consequence of this energy transfer is that the local density of the fluid decreases whilst the one of the environment does not. By the principle of Arquimedes the fluid with higher temperature and lower density is forced to go up in order to leave space for another volume of fluid whose temperature and density hasn't been modified yet. This creates a constant movement that renovates the fluid. This process is known as natural convection. In Table 1 are presented the value of the heat transfer coefficient in the main convective processes.

Otherwise if fans or pumps are used in order to make this process more efficient it is called forced convection.

Table 1. h_c value in the main convective processes.

Mechanism	Fluid	$h_c\ [W \cdot m^{-2} \cdot K^{-1}]$
Phase Transition	Boiling water	3.000-100.000
	Condensing water vapor	5.000-100.000
Free Convection	Air, gases and dry vapors	0.5-1.000
	Water and liquids	50-3.000
Forced Convection	Air, gases and dry vapors	10-1.000
	Water and liquids	50-10.000
	Liquid metals	5.000-40.000

Heat transfer by convection is more difficult to analyse than conduction transfer due to doesn't exist any single property to describe the mechanism (in conduction exists the thermal conductivity factor). Heat transfer by convection varies from situation to situation mainly due to the fluid flow conditions. Accordingly to this, in practice, analysis of heat transfer by convection is treated empirically.

Convection heat transfer depends on[6]:

- Fluid velocity
- Fluid viscosity
- Heat flux
- Surface roughness
- Type of flow (turbulent or laminar)

The basic relationship for heat transfer by convection has the same form as the heat transfer by conduction one. Some general statements about convection that makes a better understanding of the formula are:

- Heat transfer is proportional to surface area and depth of the fluid
- Heat transfer due to convection will depend on the viscosity of the fluid

It was in 1701 when Newton discovered that for forced convection of a single-phase fluid with a gradient of temperatures, the heat flux per unit area is nearly proportional to the temperature difference (between the surface and the fluid over it). This lead him to the Newton's law of cooling [7]:

$$q = h_c \cdot A \cdot \Delta T \tag{1}$$

Where,

- q : Heat flow rate by convection $[W \cdot m^{-2}]$.
- h_c : Convective heat transfer coefficient $[W \cdot m^{-2} \cdot K^{-1}]$.
- A : Surface area for heat transfer $[m^{-2}]$.
- ΔT : Temperature difference $[K]$.

h_c , as the formula shows, is dimensional. Nevertheless the traditional dimensionless form of h_c is the Nusselt number (*Nu).* This number is the ratio of convective to conductive heat transfer across the boundary under the same conditions.

$$Nu = \frac{q_{convection}}{q_{conduction}} = \frac{h_c \cdot D_h}{\lambda} \tag{2}$$

Where,

- D_h : Hydraulic diameter $[m]$.
- h_c : Convective heat transfer coefficient $[W \cdot m^{-2} \cdot K^{-1}]$.
- λ : Thermal conductivity of the bulk fluid $[W \cdot m^{-1} \cdot K^{-1}]$.

$$D_h = 4 \cdot \frac{S}{P_h} \tag{3}$$

Where,

- S : Flux section $[m^2]$.
- P_h : Dry or wet perimeter where flows the fluid $[m]$.

A Nusselt number close to one would indicate a slow motion flow, in other words, laminar flow. A large Nusselt number means a very efficient convection. It is associated with turbulent flow and can range from 100 to 1000.

Typically, for free convection, the average Nusselt number is expressed as a function of the Rayleigh number and the Prandtl number.

Otherwise, for forced convection, the Nusselt number is generally a function of the Reynolds number and the Prandtl number.

Empirical correlations for a wide variety of geometries are available that express the Nusselt number in the above forms.

6.1.1 Thermal boundary layer

When a fluid at a constant temperature (T_∞) and velocity (u_∞) flows over a flat surface in a temperature T_o appears a temperature gradient. This zone where the temperature ranges is called thermal boundary layer (TBL).

The thickness of this layer is the distance from the surface at which the temperature is 99% of the T_∞. TBL must not be confused with the hydrodynamic boundary layer (HBL) which is the distance at which the fluid velocity is 99% of the u_∞. The ratio of this two thicknesses is related with Prandtl number (see Figure 4). If the Prandtl number is 1, the two boundary layers are the same thickness. If the Prandtl number is greater than 1, the thermal boundary layer is thinner than the velocity boundary layer. If the Prandtl number is less than 1, which is the case for air at standard conditions, the thermal boundary layer is thicker than the velocity boundary layer.

$$Pr = \frac{\upsilon}{\alpha} = \frac{\mu \cdot c_p}{\lambda} \tag{4}$$

Where,

- υ : Kinematic viscosity $[m^2 \cdot s^{-1}]$.
- α: Thermal diffusivity $[m^2 \cdot s^{-1}]$.
- μ : Dynamic viscosity $[N \cdot s \cdot m^{-2}]$.
- c_p : Specific heat $[J \cdot Kg^{-1} \cdot K^{-1}]$.
- λ : Thermal conductivity of the bulk fluid $[W \cdot m^{-1} \cdot K^{-1}]$.

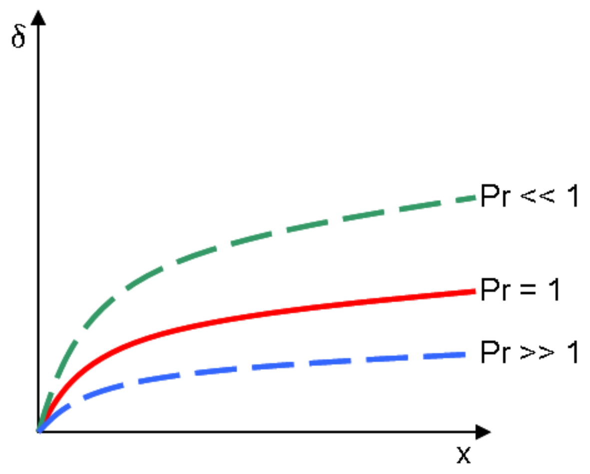

Figure 4. Thermal boundary layer thickness

6.1.2 Can we assume the convective heat transfer coefficient as non-depending temperature parameter?

It is commonly assumed that the convective heat transfer coefficient is not affected by wall temperature (instead, it is said to be dependent on the thermal driving potential). It is usually taken a single usable convective heat transfer coefficient found experimentally for every system.

However, heat transfer coefficient h_c is not a thermo-physical property but, as in the previous section is explained, it is dependent on many thermo-physical properties.

When the difference between the surface temperature of the solid and the average temperature of the fluid increases/decreases so does the value of the convective heat transfer. This happens due to its dependence with the temperature-related properties of the fluid. In plain words, when Pr changes are pronounced, h_c also drastically modifies its value (see Figure 5).

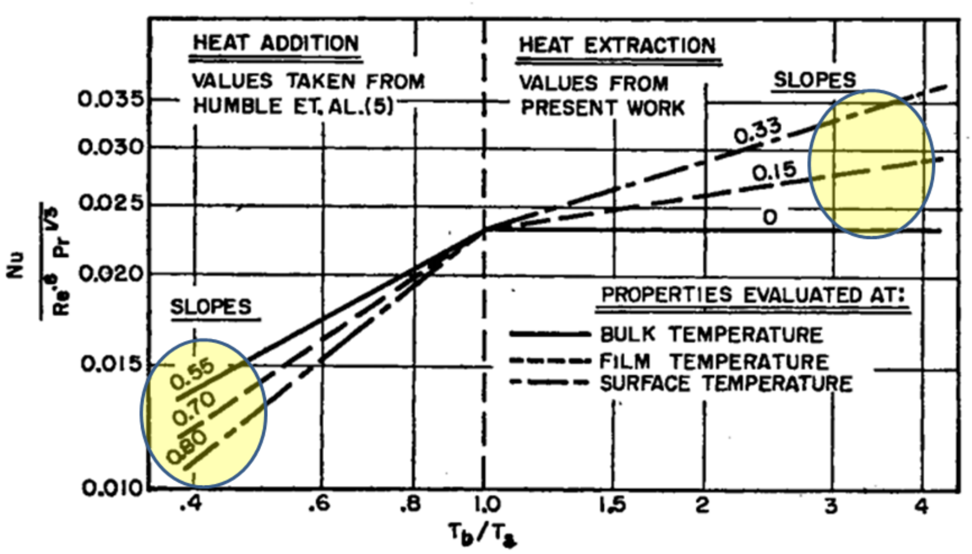

Figure 5. The effect of temperature level on heating and cooling. h_c changes due to either lower or higher $T_{bulk}/T_{surface}$ ratios [8].

So, the heat transfer coefficient is not only but also dependent on the fluid-to-wall temperature ratio as well.

6.1.3 h_c in natural convection

The thermal convection coefficient in natural convection, sometimes called free convection, can be determined using the following formulas when considering the Raspberry Pi as a flat plate [9].

- Flat surface with the hot face up:

$$h_c = 1{,}32 \cdot \left(\frac{T_{surface} - T_{bulk}}{L}\right)^{0{,}25} \tag{5}$$

- Flat surface with the hot face down:

$$h_c = 0{,}59 \cdot \left(\frac{T_{surface} - T_{bulk}}{L}\right)^{0{,}25} \tag{6}$$

Where,

- T_{bulk}: Environment temperature $[K]$.
- $T_{surface}$: Surface temperature $[K]$.
- L : Characteristic linear dimension $[m]$.

6.1.4 h_c in forced convection

For forced convection, the Nusselt number is generally a function of the Reynolds number and the Prandtl number:

$$Nu = f(Re, Pr) \tag{7}$$

$$Re = \frac{\rho v L}{\mu} = \frac{vL}{\upsilon} \tag{8}$$

Where,

- υ : Kinematic viscosity $[m^2 \cdot s^{-1}]$.
- ρ: Density of the fluid $[Kg \cdot m^3]$.
- μ : Dynamic viscosity $[N \cdot s \cdot m^{-2}]$.
- v : Maximum velocity of the object relative to the fluid $[m \cdot s]$.
- L : Characteristic linear dimension $[m]$.

However, more accurate correlations exist in order to obtain more precise results. The main aspects to consider when choosing a correlation are:

1. What is the geometry? (Flow through a pipe, around an object, over a plane…))
2. Is there a phase change?
3. What is the flow regime? (Reynolds number)
4. If the flow is laminar, is natural convection important?

Here are presented two of them for turbulent conduit flow with no phase change in a pipeline[10]:

- **Gnielinski correlation**

$$Nu_D = \frac{(f/8)\cdot(Re_D - 1000)\cdot Pr}{1 + 12{,}7\cdot(f/8)^{0{,}5}\cdot(Pr^{2/3} - 1)} \tag{9}$$

Where,

- f: Darcy friction factor that can either be obtained from the Moody diagram (as shown in Figure 6) or for smooth tubes from correlation developed by Petukhov:

$$f = (0{,}79\ln(Re_D) - 1{,}64)^{-2} \tag{10}$$

- Pr : Prandtl number.
- Re : Reynolds number.

<u>Gnielinski correlation is valid for:</u>

$0{,}5 \leq Pr \leq 2000$

$3000 \leq Re \leq 2000$

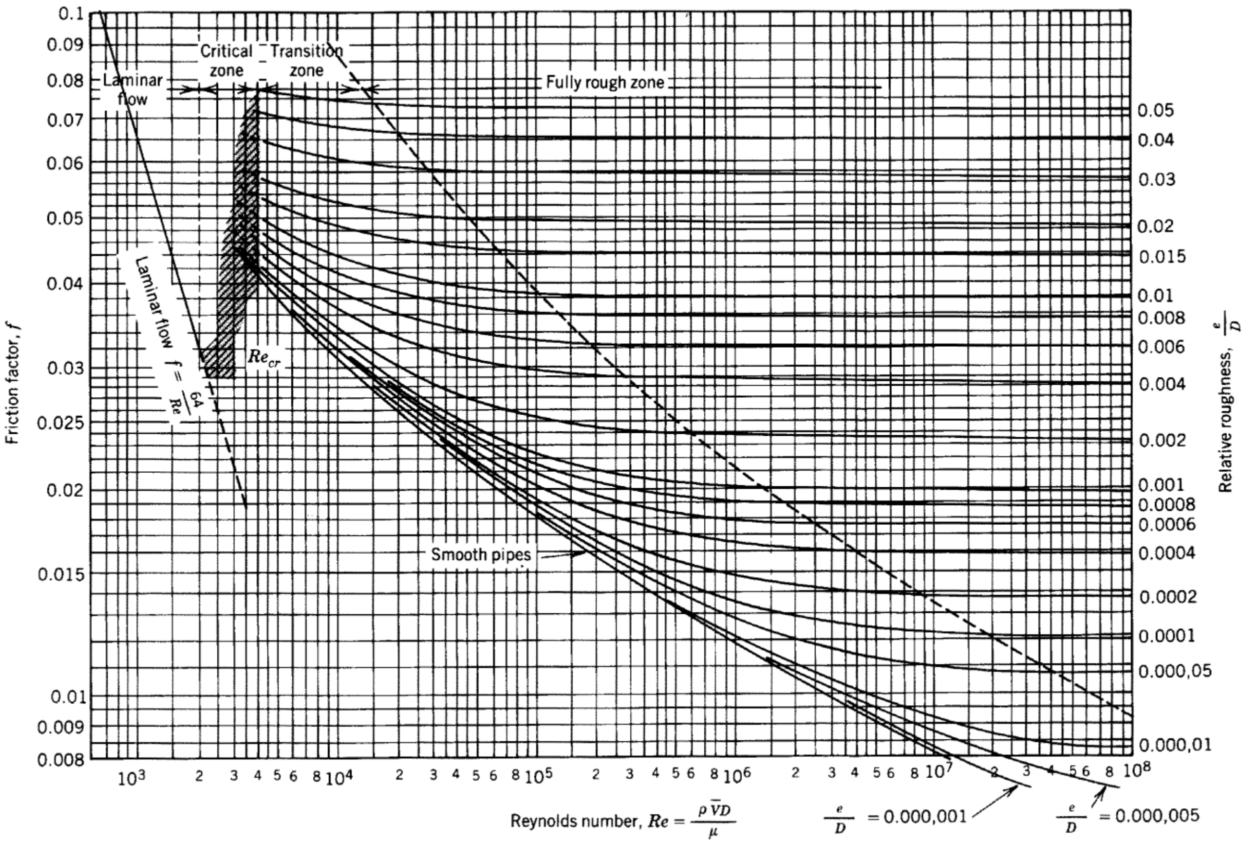

Figure 6. Moody Diagram

- **Bittus-Boelter correlation**

$$Nu = 0.023 \cdot Re^{0,8} \cdot Pr^{n} \qquad n = 0,3 \; or \; 0,4 \tag{11}$$

The exponent *"n"* depends on the service, 0,4 is used for heating and 0,3 for cooling. Heating and cooling effect the velocity profile differently because an increase of temperature modifies as well the viscosity. Heating usually makes the fluid less viscous. Otherwise, cooling, has the opposite effect, this means less heat transfer and a higher viscosity.

Exists a correction that allows you to use a direct expression in order not to consider if it's heating or cooling:

$$\emptyset_v = \left(\frac{\mu}{\mu_s}\right)^{0,14} \tag{12}$$

Where,

- μ : Dynamic bulk viscosity $[N \cdot s \cdot m^{-2}]$.
- μ_s: Dynamic wall viscosity $[N \cdot s \cdot m^{-2}]$.

Bittus-Boelter correlation is valid for:

0,7 ≤ Pr ≤ 160

Re ≤ 10.000

If this correction is added, the result is the Seider-Tate Correlation:

$$Nu = 0.023 \cdot Re^{0,8} \cdot Pr^{1/3} \cdot \emptyset_v \tag{13}$$

<u>Seider-Tate correlation is valid for:</u>

0,7 ≤ Pr ≤ 160

Re ≤ 10.000

More correction factors can be added [11]**:**

Adjust the entrance/exit consequences of short tubes (length≤10D):

$$Nu_{corrected} = Nu \cdot \left(1 + \left(\frac{D_h}{L}\right)^{2/3}\right) \tag{14}$$

And for pipe curvature:

$$Nu_{corrected} = Nu \cdot \left(1 + 3{,}5 \cdot \frac{D_h}{D_{coil}}\right) \tag{15}$$

$$D_h = 4 \cdot \frac{S}{P_h} \tag{16}$$

Where,

- D_h : Hydraulic diameter $[m]$
- D_{coil}: Curve diameter$[m]$.
- Nu : Nusselt without correction.
- L : Characteristic linear dimension $[m]$.
- S : Flux section $[m^2]$.
- P_h : Dry or wet perimeter where flows the fluid $[m]$.

When there is no pipeline but instead you have a flat plate there are other correlations. A common exception involves existence of an unheated starting length upstream of a heated section [12]. The velocity boundary layer growth begins at x=0 but the thermal boundary layer development begins at x = ξ.

The Nusselt for a turbulent flow in this situation is :

$$Nu_x = 0.0296 \cdot Re_x^{\frac{4}{5}} \cdot Pr^{\frac{1}{3}} \qquad 0{,}6 < Pr < 60 \tag{17}$$

Where,

- Re_X: Reynolds on x.
- Pr: x where boundary layer development begins.

$$Nu_x = \frac{\left. Nu_x \right|_{\xi=0}}{\left(1 - \frac{\xi}{x}^{0.9}\right)^{\frac{1}{9}}} \tag{18}$$

Where,

- $\left. Nu_x \right|_{\xi=0}$: Local Nusselt number from equation 38.
- ξ: x where boundary layer development begins.

6.2 Radiation influence

The radiation is another mechanism of heat transfer. Radiation is the transfer of heat by means of electromagnetic waves. The radiation emitted by a body is a consequence of thermal agitation of its composing molecules. Radiation heat transfer can be described by reference to the "black body".

6.2.1 Black body

The black body is defined as a body that absorbs all radiation that falls on its surface. Black bodies doesn't exist in real life.

A black body is a hypothetical body that absorbs all wavelengths of thermal radiation incident on it. They don't reflect light, and therefore appear black if their temperatures are low enough so as not to be self-luminous. All blackbodies heated to a given temperature emit thermal radiation (as shown on Figure 7).

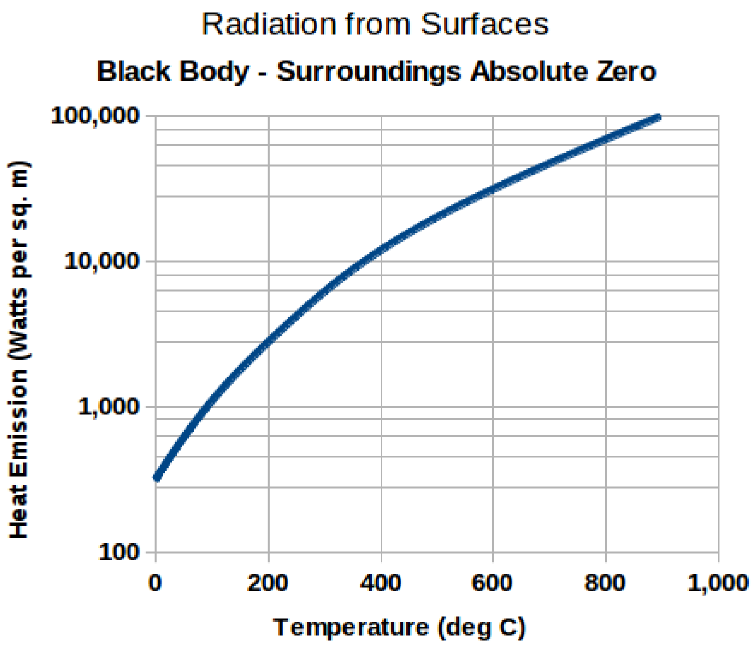

Figure 7. Black body chart of radiation by temperature

The radiation energy from a blackbody is proportional to the fourth power of the absolute temperature, surface area of the emitting body and a constant. It can be expressed with Stefan-Boltzmann Law as shown:

$$q = \sigma \cdot A \cdot T^4 \tag{19}$$

Where,

- q : Heat flow rate by radiation [W].
- σ : The Stefan-Boltzmann constant [$W \cdot m^{-2} \cdot K^{-4}$].
- A : Surface area of the emitting body [m^{-2}].

  T : Absolute temperature [K].

If the object is not ideal the Stefan-Boltzmann Law can be expressed as:

$$q = \varepsilon \cdot \sigma \cdot A \cdot T^4 \qquad (20)$$

Where,

- q : Heat flow rate by radiation $[W]$.
- ε : Emissivity coefficient of the object (it ranges from 0 to 1).
- σ : The Stefan-Boltzmann constant $[W \cdot m^{-2} \cdot K^{-4}]$.
- A : Surface area of the emitting body $[m^{-2}]$.
- T : Absolute temperature [K].

Real bodies have an emissivity that varies with temperature and frequency. In order to simplify it, bodies can be assumed as 'gray bodies' with a constant emissivity.

6.2.2 Net radiation loss rate

When a hot planar object is radiating energy to its cooler surroundings, being such surroundings very large, the net radiation heat loss rate can be expressed as equation ():

$$q = \varepsilon \cdot \sigma \cdot A \cdot \left(T_h^4 - T_c^4\right) \qquad (21)$$

Where,

- q : Heat flow rate by radiation $[W]$.
- ε : Emissivity coefficient of the object (it ranges from 0 to 1 for black bodies).
- σ : The Stefan-Boltzmann constant $[W \cdot m^{-2} \cdot K^{-4}]$.
- A : Surface area of the emitting body $[m^{-2}]$.
- T_h : Hot body absolute temperature [K].
- T_c : Cold surroundings absolute temperature [K].

7 Arduino

The role that Arduino plays in this project is essential concerning the use of NTC thermocouples. The need of a data logger for the experiments carried out along this project required a device able not only to store all the data but also to process it. Albeit there were already available data loggers at MiMT they were reserved for the practical sessions of different subjects. The limitations on the available resources lead us to find a cheap but at the same time reliable data acquisition software. And the best and more economic device that meets all this requirements is the Arduino.

This section contains a brief presentation of what is Arduino and an explanation of what does the device in order to obtain the temperature which later on will be used to define the thermic convection coefficient.

7.1 Introduction to Arduino

Arduino is an open-source prototyping platform based on easy-to-use hardware and software. Arduino can take information from his environment through his entry pins from a range of sensors. That allows you to control everything that surrounds the microcontroller (MC); lights, engines, other kind of actuators, etc. Another strong point of Arduino is that the projects can be executed without connecting it to the computer(e.g. using Bluetooth or Wifi) [13].
The hardware originally consists in a board with a microcontroller Atmel AVR and complementary components that facilitate programming and the incorporation into other circuits.
Arduino can take part in an endless range of projects. Some examples are (Figure 8):

- ✓ Electronic music
- ✓ Control Servo Motors with the Wii Mote Joystick
- ✓ Tweet-a-Watt Wireless Electricity Monitor
- ✓ Robot that Reads and Speaks RSS Feeds
- ✓ Open source Game Boy
- ✓ Controlling an RC Car with iPhone and Wii

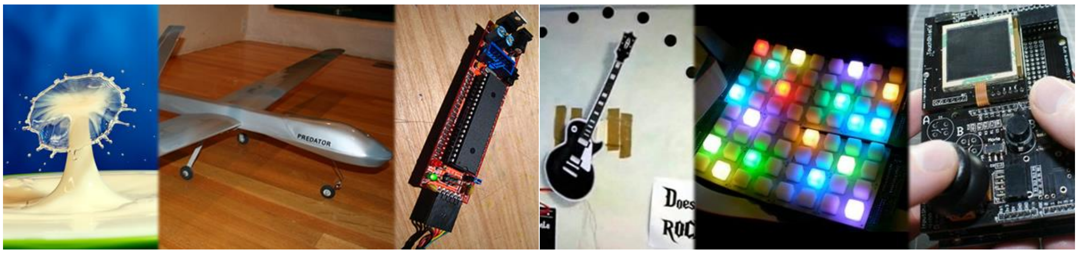

Figure 8. Examples of projects done with Arduino

On the other hand, the software. Arduino sketches can be written in a wide range of programming languages. However the Arduino project provides an integrated development environment (IDE). Its aim is to introduce programming to students and other newcomers unfamiliar with software development. It contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus [14].

Although there are several types of Arduino (Arduino Uno, Arduino Mega, Arduino Nano…) most of them have the majority of the components shown in Figure 9 in common [15]:

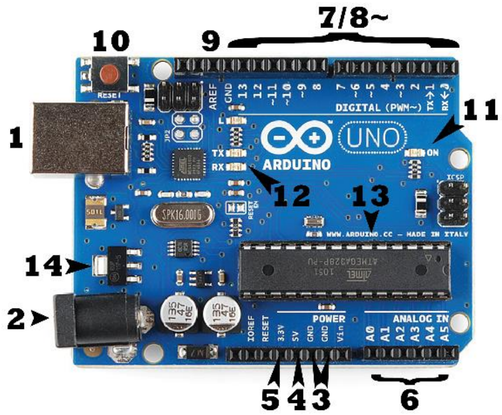

Figure 9. Common components in any Arduino [15].

1. USB connector.
2. Wall adapter power supply (barrel jack).
3. Ground pins.
4. 5 volts of power supplier.
5. 3.3 volts of power supplier.
6. From A0 to A5 are Analog In pins. They convert the analog information from the sensor to digital using an ACD.
7. From 0 to 13 are pins that can be used for both digital input (e.g. button pushed) and digital output (e.g. power a LED).
8. The pins from 0 to 13 that have this symbol (~) can be used also as PWM(Pulse-Width Modulation).
9. Analog Reference, means that you can use this pin to set an external reference voltage between 0 and 5Volts limited by the analog input pins.
10. Reset button. It connects the reset pin to ground and restart any code that is loaded on the Arduino.
11. Power led. It turns on whenever you plug the Arduino into a power source.
12. TX and RX are pins for serial communication. T stands for transmit and R stands for receive. The pins will light up at the time there is information being transmitted or received.
13. An integrated circuit (IC).
14. Voltage regulator. It allow you to regulate the amount of power that is let into the Arduino. The maximum quantity is 20 volts.

7.2 Why Arduino

Arduino unlike other data processors is not really a proper MCU in the sense that many of the functions of a MCU are hidden from the user, the whole point of Arduino is the user interface. It has an easy-to-use software for beginners but also flexible enough for advanced users. It runs on any of the three main Operating System (OS) nowadays: Windows, Mac and Linux. The main aspects that lead us to choose Arduino are[13]:

- ✓ **Development environment:** Simple and clear programming environment.
- ✓ **Cross-platform:** The Arduino Software (IDE), unlike most MCs (the majority of them are limited to Windows), can run on Windows, Mac and Linux.
- ✓ **Power consumption**
- ✓ **Cost:** Arduino boards are less expensive than the other MC platforms as we can see in the table below. The cheapest model can be bought for 30$.
- ✓ **Open source software and hardware:** This allows the users to modify or enhance the source code and also they can improve and extend the functionality of the board.

Table 2 below sums up the comparison of the main MCUs:

Table 2. Comparision of the main MCUs [16].

	Arduino Uno	Galileo	Raspberry Pi	BeagleBone Black
Cost	$	$$$$	$$	$$$
Platform	Win, OSX, Linux	Win / Liux	Win / Linux	Win / Linux
Development environment	Arduino IDE / C Variant	Arduino IDE	Linux*	Linux*
Analog Pins	6	6	None	7

*Including Python, Scratch, Perl, Java, JavaScript/Node, C, C++, and Ruby

7.3 Measuring voltage with Arduino

The IC controller installed in the Arduino UNO contains an on-board 6 channel analog-to-digital (A/D) converter. This means that it will sent an integer which will range from 0 to 1023 due to it is a 10 bit resolution [17].

Its main utility is to read analog sensors and in this particular case we want to obtain temperature from the voltage using a NTC. To measure the temperature is needed to measure the resistance in first place. However, a microcontroller does not have a resistance-meter built in. Instead, it only has a voltage reader known as analog-digital-converter as mentioned above. So what has to be done is convert the voltage into a resistance. This can be achieved by a potential divider.

The code required in order to read voltage is available in the Annex C.3. It reads the value from the ADC pin 0. This can be any number from 0 to 1023. The value sent along the serial connection is an integer value, which needs to be interpreted in order to translate it into voltage.

*More specific details of this set-up are given in the following pages.

ADCs need two points as a reference. The 5V power supply and the ground can be used (Figure 10 and Figure 11). Hence, a reading of 561.5 in the ADC represents a potential of 2.5V. Converting an analogic value to a potential can be performed using the following equation:

$$V_{out} = \frac{x}{1024} * V_{in} \tag{22}$$

Where,

- V_{in} : Power supply voltage [V].
- x : Integer value ranged from 0 to 1023. This means 1024 units.
- V_{out} : Output voltage [V].

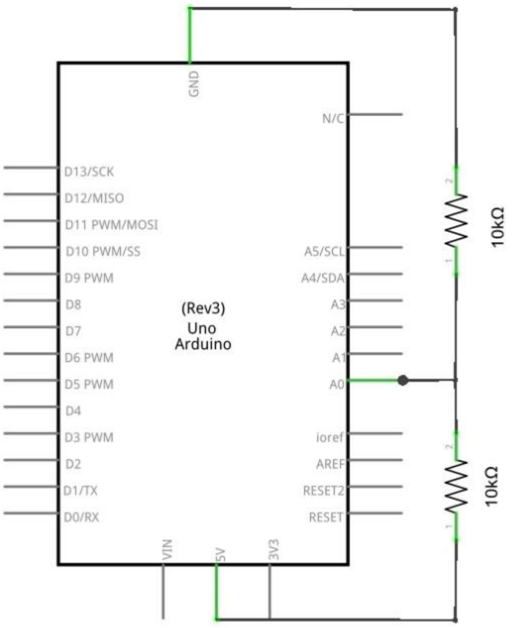

Figure 10. Schematic of the potential divider.

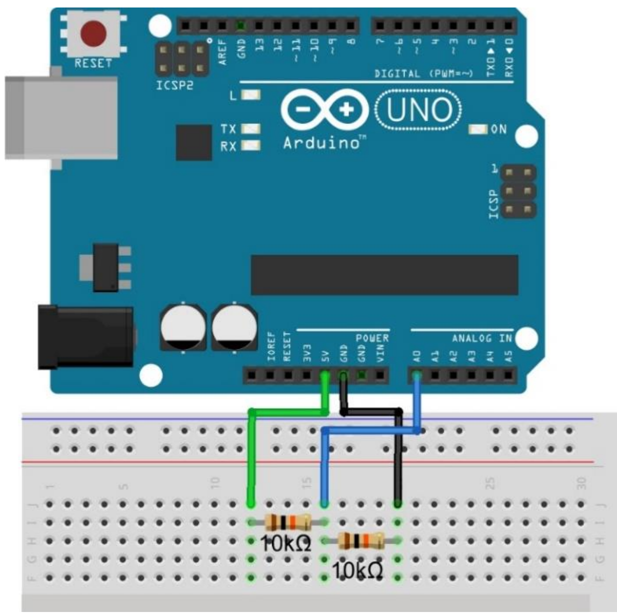

Figure 11. Potential divider using Arduino.

In Annex B.1 and Annex B.2 there is some extra information about Arduinos integrated development environment and some explanations about the main commands it has as well.

7.4 From resistance to temperature

Once how to obtain the voltage from the Arduino is known, the next step is to learn how to convert it to resistance in order to obtain the temperature of the NTC, which is the real interest. This question can be solved via a potential divider. It depends on your objective to select a higher or a lower value for the resistance. In this case a 10kOhm resistor has been selected. The equations below show the process carried out by the Arduino to obtain the resistance in ohms [18]:

From the potential divider,

$$V_{Out} = \left(\frac{R}{R + 10000}\right) * V_{in} \tag{23}$$

Now if it is connected up to Arduino a number will be performed. So mixing the equations (1) and (2) it results the following expression,

$$x = \left(\frac{R}{R + 10000}\right) * 1024 \tag{24}$$

It looks nice because it is non dependant on the voltage used. A little math is required to put the R on the left instead of x,

$$R = \frac{10000}{\frac{1024}{x} - 1} \tag{25}$$

Where,

- V_{in} : Power supply voltage [V].
- x : Integer value ranged from 0 to 1023. This means 1024 units.
- R : NTC variable resistor [Ω].
- V_{out} : Output voltage [V].

But the real interest is the temperature. To obtain it from the resistance in a thermistor NTC two methods exist:

Steinhart-Hart method

The equation that follows permits to obtain the relations between temperature and resistance

$$\frac{1}{T} = A + Bln(R) + C(ln(R))^3 \tag{26}$$

Where:

- T : Temperature [K].
- R : Resistance at T [Ω].
- A, B, C : Steinhart-Hart coefficients.

There seems to be an universal value for each one of the 3 Steinhart-Hart parameter as shown in Table 3. However in this project this coefficients have been calculated experimentally. The process of obtaining these and its values are shown in the next section.

Table 3. Universal values that work for most NTC thermistors.

A	B	C
0.001129148	0.000234125	8.76741E-0.8

The Figure 12 shows a possible setup for determining the parameters of the Steinhart-Hart equation.

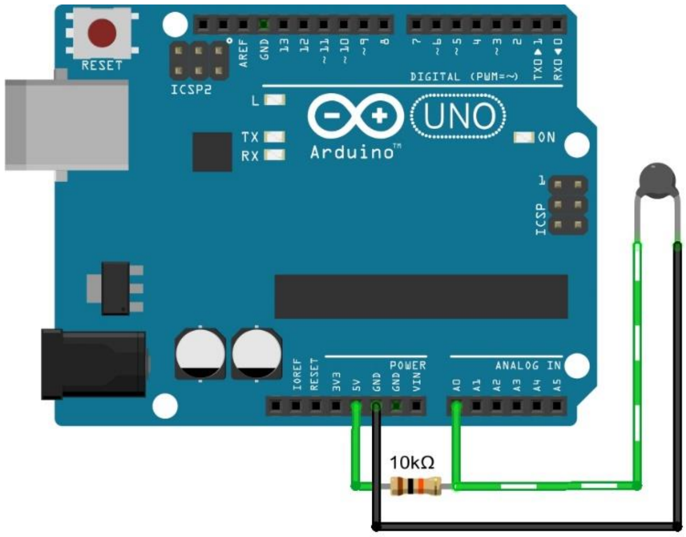

Figure 12. Breadboard example of the Steinhart-Hart method.

Beta Factor method

$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{\beta} ln\left(\frac{R}{R_0}\right) \tag{27}$$

Where,

- T_0 : Reference temperature of the thermistor. Usually 298.15K [K].
- R_0 : Resistance at T_0 [Ω].
- β : Material constant [K].
- T : Temperature [K].
- R : Resistance [Ω].

The Figure 13 shows a possible setup for determining the parameters of the Beta Factor method.

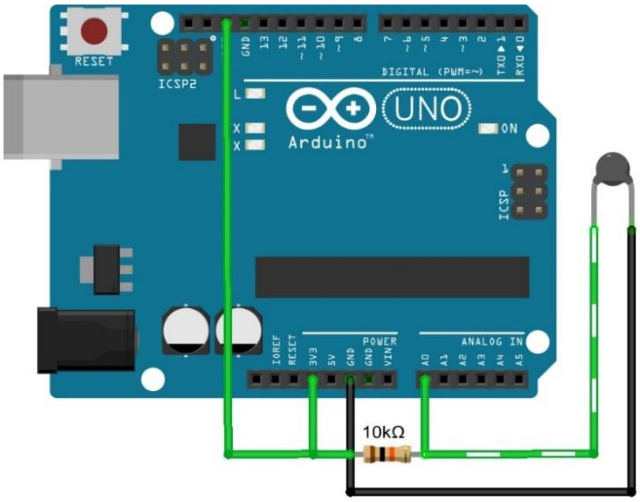

Figure 13. Breadboard example of the Beta factor method.

There is just one small difference between the two methods in terms of circuit assembly. It is the power supply. In fact that should even be considered a difference because it can be connected either in the 3.3V pin or in the 5V pin. The difference between both of them is that the first image of the board the reading goes through a secondary reading and reduces the noise [18]. It depends on your objective to select a higher or a lower value for the resistance. In this particular example it's 10kOhm.

All lines of code are available at Annex C.3.

7.5 Explanation of the Steinhart-Hart and Beta equations

The need to find a relation between resistance and temperature lead the scientific community to develop a model to explain that [19]. They assumed that the thermistor materials followed the model in conductivity known as intrinsic conduction, which is represented by this equation:

$$\ln R = \mathrm{A} + \frac{\beta}{T} \tag{28}$$

Where,

- T : Absolute temperature [K].
- β : Material constant [K].
- R : Resistance [Ω].

This expression can be adapted to the NTC thermistor adding R and T at a Standard reference temperature (typically 298,15 K),

$$\ln R_0 = \mathrm{A} + \frac{\beta}{T_0} \tag{29}$$

and by solving the system composed by equation 28 and equation 29 results the following expression:

$$\ln R - \ln R_{T0} = \ln \frac{R}{R_{T0}} = \beta * \left(\frac{1}{T} - \frac{1}{T_0}\right) \tag{30}$$

Which equals to

$$T = \left(\frac{1}{\beta} * \ln \frac{R}{R_{T0}} + \frac{1}{T_0}\right)^{-1} \tag{31}$$

Where,

- T : Absolute temperature [K].
- β : Material constant [K].
- R : Resistance [Ω].
- R_{T0} : Resistance at the reference temperature [Ω].
- T_0 : Reference temperature. Usually 298.15K [K].

Therefore, if two points or a single point and the β factor are known, the equation can be solved easily. This was called the Beta Factor Method.

Unfortunately, this equation seems to be valid only for applications requiring narrow temperature spans of 20°C or less. Nowadays the technology is far more precise and for those applications requiring temperature spans of 50°C or more another model is needed. At the beginning the new improved resistance/temperature curve fit models pretended to take into account the temperature dependence on Beta, but solutions were more difficult and not accepted for commercial thermistor applications.

In their search for an empirical expression that could provide more accurate results Steinhart and Hart found the following equation while searching for an accurate interpolation of the resistance/temperature characteristic in the oceanographic temperature range of -2°C to 35°C [20]:

$$\frac{1}{T} = A + B * (\ln R)^2 + C * (\ln R)^3 \tag{32}$$

Later on was discovered that by eliminating the squared term the equation improved the curve-fitting results for its oceanographic tests. The following expression is the form that the final equation adopted. This one is the model that is known as the Steinhart-Hart Equation.

$$\frac{1}{T} = A + B * \ln R + C * (\ln R)^3 \tag{33}$$

Where,

- T : Temperature [K].
- R : Resistance at T [Ω].
- A, B, C : Steinhart-Hart coefficients.

8 Calibration of the Thermistor

This section covers the NTC calibration experiment. It includes a brief presentation of the different kinds of thermistors, the setup of the computer and the steps carried out in order to obtain the data needed to perform the calculations in order to obtain the experimental coefficients of the thermistor.

8.1 Types of thermistor

Thermistors are ceramic or polymer-based devices used to measure temperature. The resistance of a thermistor varies with temperature in a non-linear way, so by measuring resistance, one can infer the temperature of the environment or any object in thermal contact with the thermistor. Thermistors are relatively inexpensive and are available in a variety of physical configurations.

There are two big kinds of thermistors:

- **Negative Temperature Coefficient (NTC):**

  In NTC thermistors when the temperatures rises the resistance diminishes. This equals to say that the current increases its value.

- **Positive Temperature Coefficient (PTC):**

  On the other hand a PTC also changes its resistance when the temperature increases. However, in this case both of them rise or fall at the same time. Its main characteristic though, is that when they overpass a certain temperature called, Curie's Temperature, behave like an NTC [21].

NTCs, unlike PTC thermistors, are used when there is a constant change in the resistance value in a wide range of temperatures. They offer mechanic, thermic, and electric stability apart from a high degree of sensitivity.

PTCs are commonly used when a drastic change on the resistance in a specific temperature or intensity is required. They can be found for example in a circuit protection system as replacements for fuses. Whilst the fuses are only usable one single time the PTCs have the characteristic that, when the device where are they mounted heats up, the resistance increase and consequently limits the current as shown in Figure 14. When the device cools down so does the resistor and consequently the currents flow again.

In our case study it has been used a Rohs BC thermistor NTC 10kOhm.

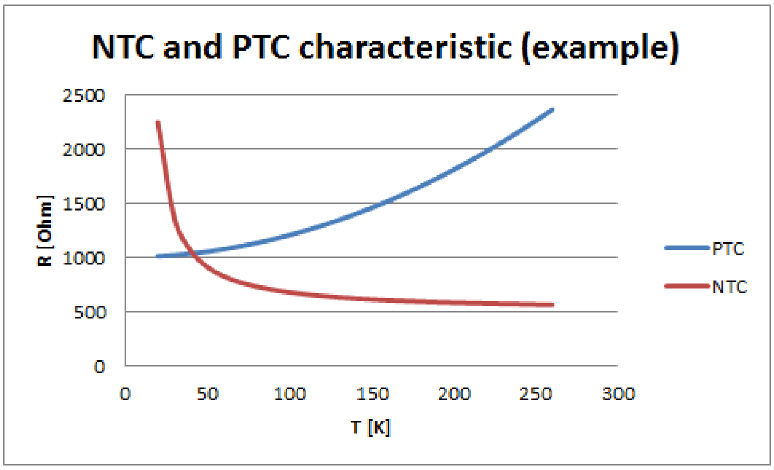

Figure 14. NTC and PTC comparison chart

8.2 Boot up the program

The first step was to install the IDE software from the main Arduino's web page. At the beginning it was not easy to understand how it works but after a few practice and some tests we acknowledged the main features that this microcomputer can offer and were capable of design our experiment.

Once we had analysed the different features and options the software could offer us we began with the calibration experiment.

The next step was to configure the serial port. This was really important cause is how we can connect the Arduino with our computer. In order to do that we simply had to plug in an USB wire from the computer to the Arduino's port. The program automatically detected our device (COM3). This, which seems pretty easy in our case, is particularly important when you have more than one board because you must select in the interface which one you want to use. Once the connections were settled we had to run the sketch that allowed us to obtain the resistance. We also decided to print in the layout the V_{out} potential and the temperature with the universal coefficients as well. We did this basically to compare the temperature between the type K thermocouple and the NTC using the standardized coefficients. The lines of code are available on Annex C.3.

8.3 Prepare and run the experiment

The materials needed for the calibration were:

- Arduino Uno
- NTC
- 10kOhm resistance
- Wire
- Chronometer
- Oven, model HK50
- Computer to run the Arduino and visualize the data
- UniCal TC+ multifunction indicator-simulator

Once we had the program boot up we had to set up the calibration (see Figure 15 and Figure 16). We used a controlled oven and a calibrated temperature sensor type K (the certificate of both type K sensor and the oven thermometer are available in Annex C.8) in order to be able to assign a value of temperature for each resistance.

The steps followed to prepare and execute the calibration were:

1. Connect the potential divider (10kOhm resistor + NTC) with the Arduino.
2. Cover the NTC terminals with an insulating material. In our case we used Teflon.
3. Place the NTC thermistor and the temperature sensor together as close as possible from one and other in the oven.
4. Fill the space in between the two sensor with magnesium oxide in order to reduce the empty areas and heat loss.
5. Start the oven. Record the voltage and resistance from the NTC and the temperature from the sensor.
6. Increase five degrees the oven temperature every 10 minutes and write down the temperature read from the type K and the resistance and voltage from the NTC . The experiment starting point was at 20°C and its last one was at 80°C.

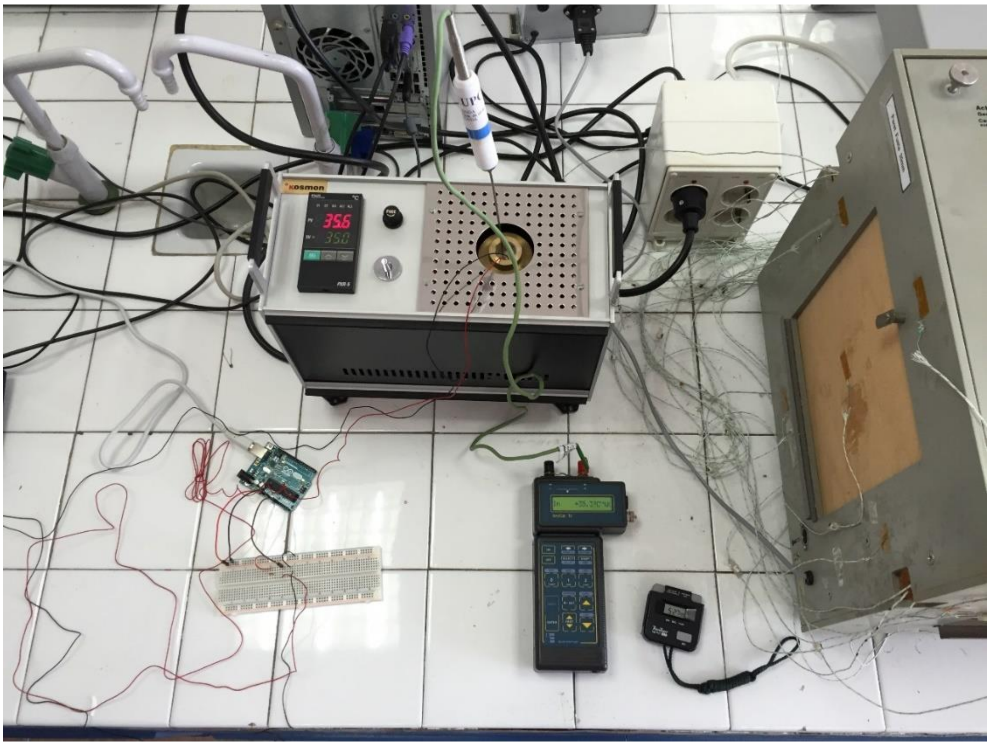

Figure 15. Assembly of the calibration experiment.

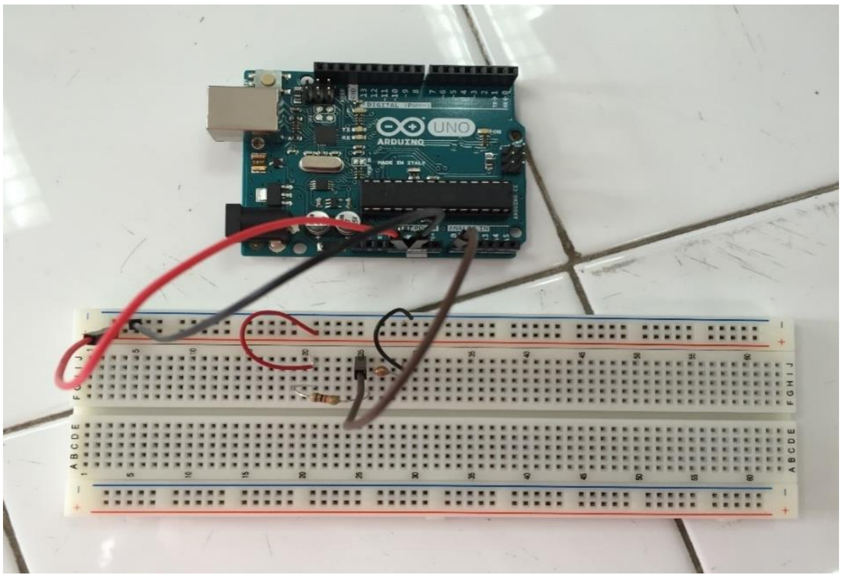

Figure 16. A more precis view of the potential divider.

In Table 4 are the values of the resistance and temperature measured (more data of this experiment can be found at Annex C.5).

Table 4. Calibration data for a Rohs BC thermistor NTC 10kOhm at 19.3 °C. Data recorded with an UniCal TC+ multifunction indicator-simulator and a Sensor type K.

Resistance (Ohms)	Sensor temperature (°C)
11244	21,7
9806	25,2
8028	30,6
6704	35,6
5585	40,7
4712	45,6
3989	50,6
3385	55,5
2880	60,7
2472	65,6
2147	70,5
1838	75,6
1596	80,5

8.4 Obtaining the equation

Two different equations were available at this point. The Steinhart-Hart method or the Beta Factor method. We ended up using the first one due to the higher level of precision.

In order to find the parameters of the equation (33) we needed a program to process all that data registered from the Arduino. To do that we chose Matlab, cause it was the one we were familiar with , but other could have been used. The lines of code and an explanation of how to obtain the resistance from the temperatures are available in Annex C.4 and Annex C.1 respectively. The Table 5 shows the values obtained.

Table 5. Values of the coefficients obtained after the calibration

Coefficient	Value
A	8.482012645e-04
B	2.611146625e-04
C	1.328876153e-07

$$\frac{1}{T} = 8.482\mathrm{e}-04 + 2.611\mathrm{e}-04 * \ln R + 1.329\mathrm{e}-07 * (\ln R)^3 \qquad (34)$$

We can also obtain different charts. Enclosed in the Annex C.6 there are the inverse model and the degree 3 polynomial approximation charts. The following Figure 17 is the Steinhart-Hart model.

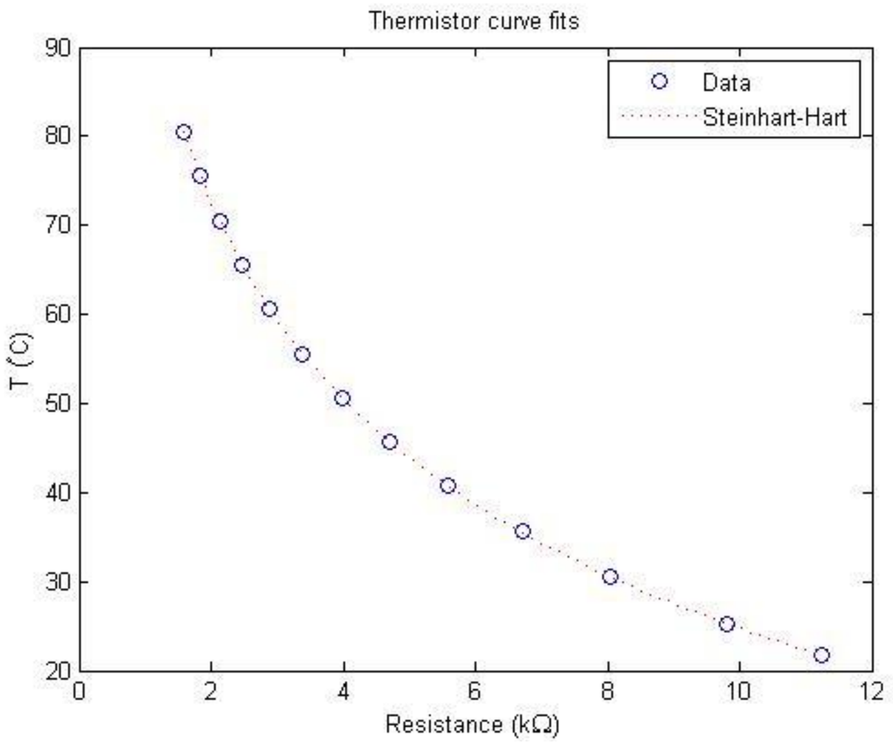

Figure 17. Thermistor curve fit.

8.5 Verification

Once the calibration was done a verification was compulsory. In order to do that is necessary to compare a range, as wide as possible, of different temperatures. The university thermal engineering laboratory has a small-sized model house with replaceable side walls used for determining the heat transfer coefficients (h_c) of various walls and windows and for establishing the heat conductivities of different materials (see Figure 18). In order to achieve this purpose the temperatures on the inside and outside of the walls are measured at a constant interior and outer air temperature (in the steady state). In each wall, specified on the Table 6, there is placed a type K thermocouple and this is connected to a data logger which in turn is connected to the computer.

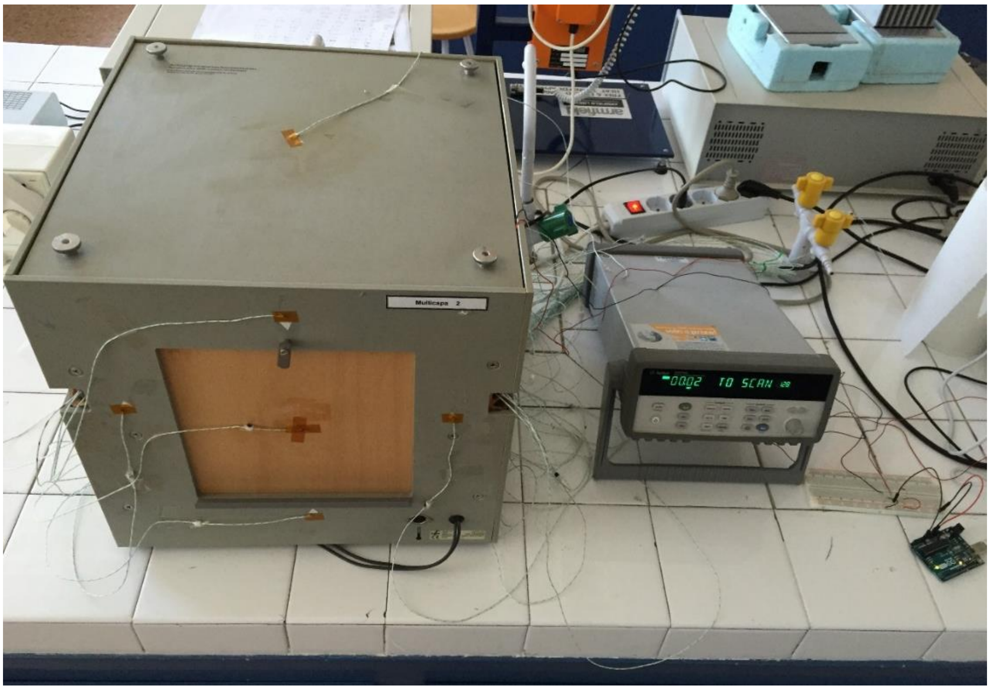

Figure 18. Experimental set-up of the thermal house

The verification experiment consisted in to set as close as possible the NTC and the type K in the different walls/materials. Then we waited about 10-15 minutes to let the components stabilize and read the right temperature. Once that time was past we took the NTC and placed it into another wall and waited for another 10-15 min. We took 3 measures for each wall/material and then we calculated the average value of the temperatures in order to be more precise. However in the following Table 6 only the average of all of them is shown. The whole data of the verification can be found on the Annex C.7.

Table 6. Comparison between the type K thermocouple and the NTC. Lab. Temperature: 22,4°C.

	Type K thermocouple [°C]	NTC [°C]	Absolute difference [°C]
Environment (Outside)	26,87	27,17	0,30
Polystyrene (Outside)	29,91	30,52	0,61
Multilayer (Outside)	30,1	30,18	0,08
Wood (Outside)	36,09	36,71	0,62
Glass (Outside)	43,47	41,95	1,52
Roof (Outside)	30,24	30,71	0,47
Multilayer (Inside)	55,91	55,42	0,49
Wood (Inside)	52,41	53,08	0,67
Glass (Inside)	46,58	46,77	0,19
Environment (Inside)	58,15	57,92	0,23
AVERAGE			**0,52**

As we can see the average difference between our NTC and the type K thermocouple is 0.52°C. We have considered this difference small enough to take for good our calibration. The main aspects that could have led to this half degree of dispersion within the thermometers (type K and NTC) is due to the following reasons:

1. In each wall/material we had a different thermocouple. Although all are the same type and model there are slightly differences between one another that may cause a dispersion of a centigrade.
2. Even though we tried to put as close as possible the NTC and the thermocouple it was fiscally impossible to place them in the same spot. This cause small differences on the measurements because the distribution of temperatures might be different in less than 1mm in certain materials.
3. The uncertainty of our NTC thermistor.
4. Every type K sensor has its own uncertainty meaning that it is almost impossible that they give us the same measurement.

However, as the previous table shows, the glass seems to present a higher difference in comparison of the other materials. This led us to the following two conclusions:

- ✓ The Figure 17 indicates that from 40°C and above the temperature rises very quickly when the resistance varies very little. This means an increase of the sensibility of the NTC in terms of resistance.
- ✓ The fact that this only happens with the glass and does not happen with the inside temperatures is due to an insulating problem.

  The difference between the environment temperature (22,4 °C) and the one of the glass (43,47°C) is very high. This means that a poor insulation will lead the NTC to measure bad values because it will read the temperature from the air and also from the glass.

  The NTC must be very well insulated in order to only measure the surface temperature of the glass otherwise will also read the value from the air temperature. Moreover if we take a look into the inside environment temperature according to the first point the temperature should vary very quickly with little variation of the resistance but it was stable. This is due to the NTC was fully surrounded by air at more or less the same temperature and also well insulated with the thermal house.

The Table 6 shows a very little difference in terms of temperature between the type K and the NTC for any of the inside temperatures. This seems to be contradictory against what it's said before but it is not. The difference between the inside type K and the inside environment temperatures is not as high as the outside one. This conducts the NTC to read a more precis value as the results uphold but in fact it is still bad insulated.

The Annex C.2 contains information about why has been selected two decimals in terms of temperature if the Arduino could provide with more.

8.6 Uncertainty

Uncertainty of measurement, in plain words, is the doubt that exists about the result of any measurement. Any person might think that well-made rulers, clocks and thermometers should be trustworthy, and give the right answers. But for every measurement - even the most careful - there is always a margin of doubt [22].
In every calibration and verification the uncertainty of measurement must be reported on the certificate. In our case a calibration of a NTC thermistor has been performed. The standard formula to calculate the uncertainty is as follows [23]:

$$u_T = \sqrt{{u_s}^2 + {u_C}^2 + {u_P}^2 + {u_R}^2} \tag{35}$$

Where,

- u_T : Uncertainty of the calibration [°C].
- u_s : Standard deviation factor [°C].
- u_C : Correction factor [°C].
- u_P : Pattern uncertainty [°C].
- u_R : Equipment resolution [°C].

$\boldsymbol{u_s}$ **:**

$$s = \sqrt{\frac{\sum(x_i - \bar{x})^2}{n-1}} \tag{36}$$

$$u_s = \frac{s}{\sqrt{n}} \tag{37}$$

Where,

- n : Number of observations.
- x_i: Observations.
- $\bar{x}$: Mean of the observations.
- s : Standard deviation of the observations.

$\boldsymbol{u_C}$ **:**

$$u_C = \frac{C_p}{\sqrt{3}} \tag{38}$$

Where,

- C_p : Difference between the pattern observation at a certain temperature and the average value of the same temperature observation from the thermistor NTC.

$\boldsymbol{u_I}$ **:**

$$u_I = \frac{I_p}{k} \tag{39}$$

Where,

- I_p : Uncertainty of the pattern. In our case it has a different value depending on the temperature. The certification of the calibration pattern can be found at Annex C.8. An uncertainty of 0,4°C has been settled for the measurement of a temperature ranging from 12,5 °C to 37,5 °C. The other temperatures have been assigned a 0,3 °C of uncertainty.
- k : Coverage factor. When k =2 it provides a level of confidence of approximately 95%.

$\boldsymbol{u_R}$ **:**

$$u_R = \frac{R_p}{\sqrt{3}} \tag{40}$$

Where,

- R_p : Resolution of the thermistor. The NTC thermistor used with the Arduino as a data logger provided us with four decimals. However only two decimals has been taken due to the real capacity of the component.

The value of the uncertainty has been chosen as the maximum of each measurement. It is done this way in order to secure the calibration itself and also because it doesn't make sense assign a specific uncertainty according to a certain range of temperatures as shows the data sheet. In the Annex C.7 there are all the data and the calculations carried out in order to obtain the uncertainty for each measurement.

The uncertainty that has been picked up for this project is:

$$u_T = 0{,}89\ °C$$

9 Raspberry Pi

Raspberry Pi 2 meets the requirements to work as a computer cluster. It can be connected in parallel with other devices in order to develop a specific function. It has been chosen this device instead of another model due to its economic price and its fast and good data processor.

In this section Raspberry Pi 2 is presented in the first instance. There is also an experimental part which includes: detection of the hot spots, measure of the heat generation and calculation of the thermal convection coefficient.

9.1 Brief introduction to Raspberry Pi

It's a small credit-card-sized single board computer. It is intended to help people learn more about programming and how the computer works. In fact it was developed in United Kingdom by the Raspberry Pi Foundation to promote the teaching of basic computer science in school and third world countries.

Several generations of Raspberry Pi's have already been released. The first one, the Pi 1, went on sale in February 2012 in two models A and B (a more specific one). Later on A+ and B+ were released. The model used in this project, the Raspberry Pi 2, was released in 2015 [24]. All models can be easily used by Raspbian, the most downloaded OS for the Raspberry Pi users. Figure 19 shows the main components of a Raspberry Pi 2

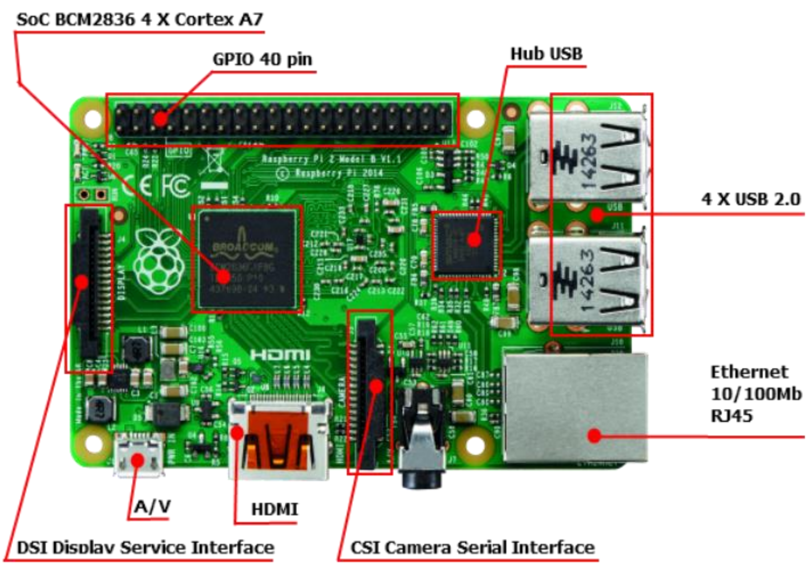

Figure 19. Raspberry Pi 2 main components.

9.2 Why Raspberry?

Raspberry Pi 2 has been chosen for this project but no only due to its cheap price but also for[25]:

- ✓ Low cost device.
- ✓ Easy to use.
- ✓ Based on an open source from Linux meaning that anyone can have free access to the software.
- ✓ Great connectivity.
- ✓ Versatile device. The user can play games, listen music, watch videos, make a security camera, program your tasks…
- ✓ Fast launch.
- ✓ The ARM Cortex-A7 processor works well with the last GNU/LINUX distributions such as Ubuntu and Windows 10.

9.3 Hot Spots Detection

In order to thermally characterize the Raspberry Pi 2, the components that generate more heat must be identified. Since the temperature reached by these critical components is expected to differ considerably from the laboratory ambient temperature, the NTC thermistor is not suitable to be used. Thus a calibrated Type K thermocouple has been used.

Previously to start the experiment some extra tasks had to be done. In order to keep the working space as clean as possible it was determined to use a Secure Shell network protocol. It was used a free open-source terminal emulator, serial console and network file transfer application called PuTTY. This application allowed us to use the Raspberry Pi 2 from another computer without the need of being connected with a wire.

The next step was to search for a test in order to force the Raspberry Pi use a 100% of its CPU capacity to detect the maximum temperature that can reach the board whilst working and see as well if it can keep that temperature in a stationary value.

At this point to different tests were considered:

- ➢ **Benchmark test:** This tests are a set of programs that allows the user to determine the performance characteristics of the device. The benchmark offers a wide range of different tests depending on what you want to study (CPU, memory, graphs, reading…)

- ➢ **Stress test:** These programs have command line parameters that determine running time and generally include performance measurements, reported at regular intervals, to identify speed reductions due to such as overheating or system interference.

The both tests were executed in order to compare different performances of the device. The results however were very similar.

The command lines used in order to run the tests were:

Benchmark

- *nohup sysbench --num-threads=8 --test=cpu --cpu-max-prime= 10000000000 run & watch sudo cat /sys/devices/system/cpu/cpu0/cpufreq /cpuinfo_cur_freq*
- *sudo sysbench --test=cpu --num-threads=4 --cpu-max-prime=20000000 --max-time=0 run*

Stress

- *stress --cpu 8 --io 4 --vm 2 --vm-bytes 128M --timeout 3600s*

There is no substantially difference between the two benchmark tests. However in the “nohup” it has been added the command “watch”. This command shows the actual frequency in which the device is working and it was used to confirm that the Pi was running at his full speed. Some users have been reporting temperature highs of just 45°C when running the benchmark, which is well below what it should be (around 60 °C). The issue appears to be that certain firmware/kernel versions are failing to scale the processor correctly, meaning that the BCM2837 is spending its time at 450MHz instead of 900MHz even when the benchmark is running - killing performance but keeping the temperature down.

On the other hand, to measure the temperature on the board the Type K thermocouple was used (see Figure 20 and Figure 21). The NTC was discarded due to the range of the surface temperatures we wanted to measure. It was used to measure the environment temperature. The fact that most of them were above 40 °C and the lack of resources to perform a well insulation tied down us to use the type K.

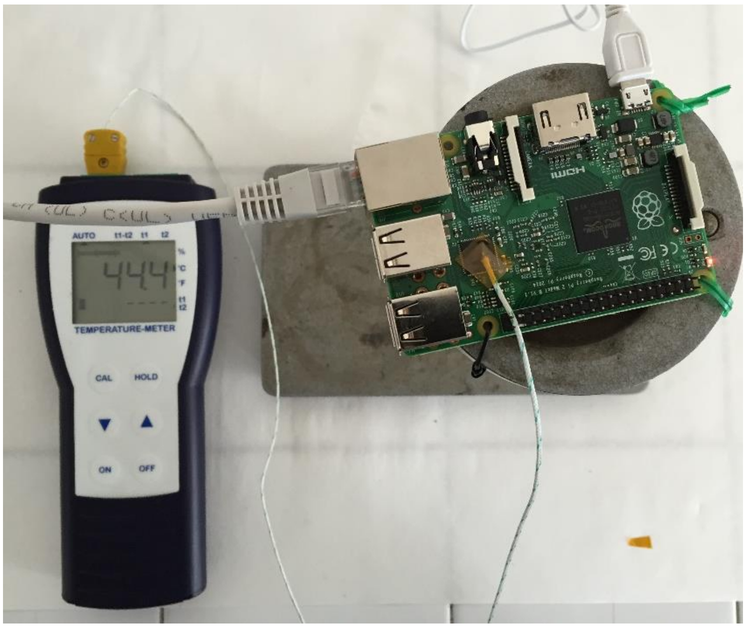

Figure 20. Experimental set up with the type K thermocouple (front face).

Figure 21. Experimental set up with the type K thermocouple (rear face).

The following Figure 22 shows the expected thermal image of a Raspberry Pi 2 whilst running a test. This image was also used to corroborate the experimental results.

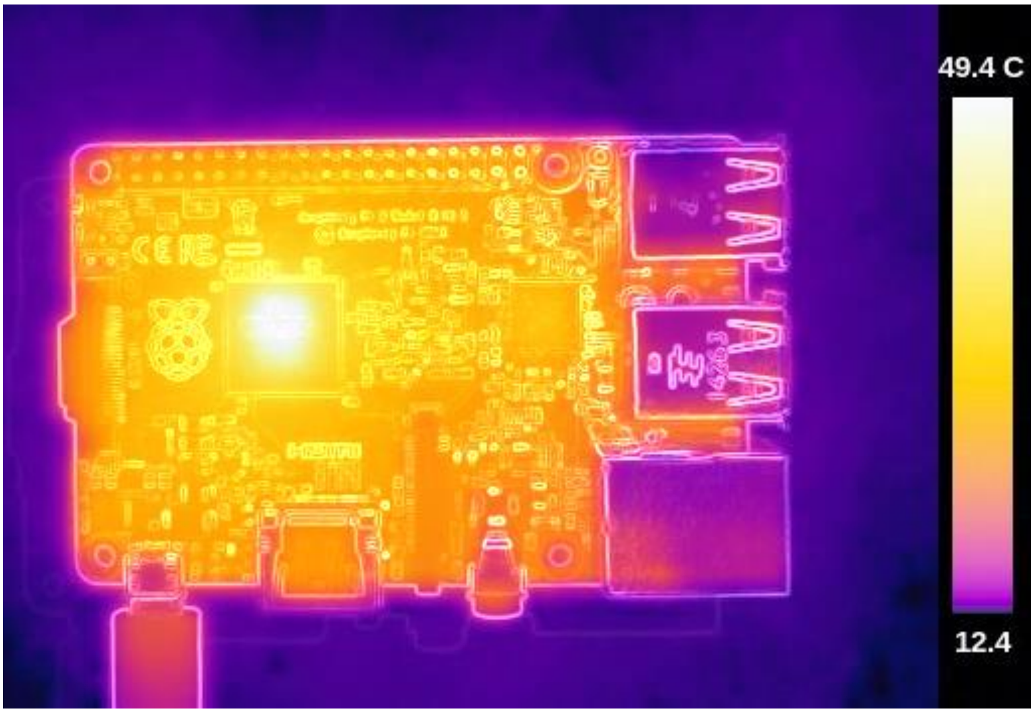

Figure 22. Thermal image whilst running a benchmark test [26].

The test had to be executed at least 15 minutes before the temperature measurements due to it was assumed that during this time the board had enough time to reach the steady state. In fact it was experimentally checked that when the test is run the device takes less than one minute in achieve the 60 °C. However some extra minutes were given in order to see if the device was increasing his temperature or not. This was also verified whilst taking the measurements with the type K thermocouple. The value of the temperature was stable, it was neither increasing nor decreasing. The temperatures were taken from the same spots in the front and rear face of the Raspberry Pi 2 (Figure 23 and Figure 24). This points were selected according to the temperature distribution of the thermal image. We selected those components with the highest temperature and also the Bakelite. The dimensions of those components and the Bakelite are shown in Table 7.

Table 7. Dimensions of the spots selected

Zone	Component	Dimensions [cm]
1	Bakelite	8,72x5,65
2	Processor	1,4x1,4
3	USB hub	0,9x0,9
4	RAM	1,2x1,2

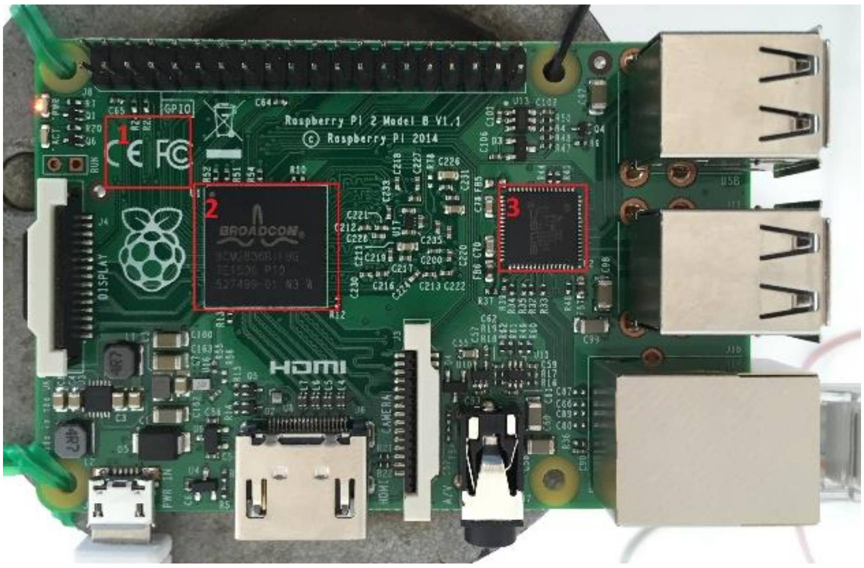

Figure 23. Surfaces where the temperature was measured (front face).

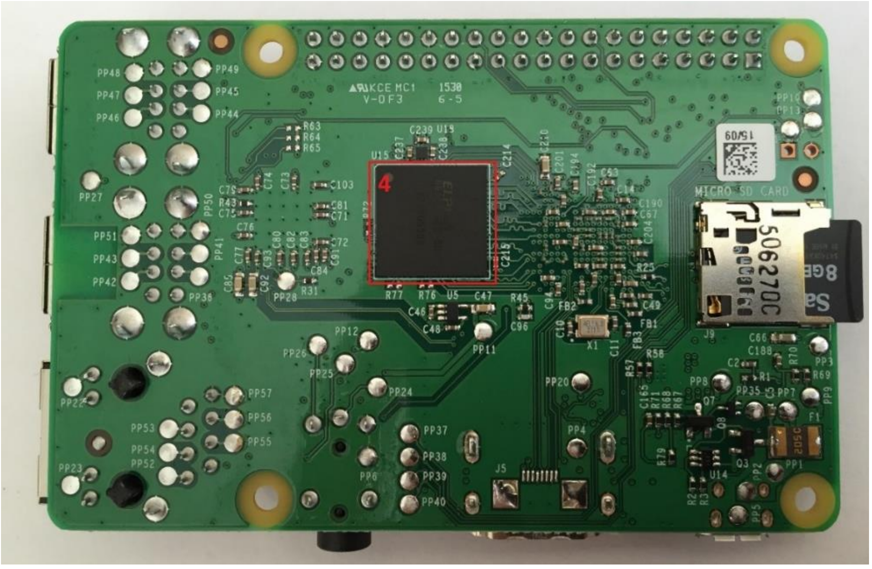

Figure 24. Surfaces where the temperature was measured (rear face).

To provide more accurate results three measurements were taken per each spot in order to perform more realistic calculations in the further sections. Also this experiment was realized two more times to prove more trustworthy results. In each of them were run the two other tests left mentioned in the previous section. The results in Table 8 and Table 9 are the ones of the third and last experiment. The data of the other two experiments can be found in Annex D.2.

The command used for this third one was:

sudo sysbench --test=cpu --num-threads=4 --cpu-max-prime=20000000 --max-time=0 run

Table 8. Front face temperature measurments.

Front face					
Zone	**Air [°C]**	**1st measure [°C]**	**2nd measure [°C]**	**3rd measure [°C]**	**Average [°C]**
1	23,2	42,4	41,7	41,1	41,7
2	23,2	54,3	57,8	54,1	55,4
3	23,2	45,9	46,3	45,2	45,8
4	23,2	42,8	42,4	41,9	42,4

Table 9. Rear face temperature measurments.

Rear face					
Zone	**Air [°C]**	**1st measure [°C]**	**2nd measure [°C]**	**3rd measure [°C]**	**Average [°C]**
1	23,2	40,9	39,8	42,5	41,1
2	23,2	49,9	52,5	50,5	51,0
3	23,2	43,2	44,5	43,7	43,8
4	23,2	45,1	45,3	44,9	45,1

This temperature values are used for the power generation. In order to do that the average temperature of those three measurements has been calculated. It has been assumed the average value as the whole component temperature. The same was done with the Bakelite.

The exact same test has been performed with fins. The temperatures have been taken from the same zones. Three sets of fins were acquired and mounted one in the microprocessor, one in the RAM and one in the USB hub. The fins were stick to the Raspberry Pi with a special glue (see Figure 25, Figure 26 and Figure 27). The Table 10 and Table 11 show the results of the test with fins.

Table 10. Front face temperature measurments with fins.

Front face					
Zone	**Air [°C]**	**1st measure [°C]**	**2nd measure [°C]**	**3rd measure [°C]**	**Average [°C]**
1	22,9	40,9	40,6	40,7	40,7
2	22,9	51,7	52,4	51,8	52,0
3	22,9	45,1	45,6	45,3	45,3
4	22,9	41,3	41,5	41,1	41,3

Table 11. Rear face temperature measurments with fins.

Rear face					
Zone	**Air [°C]**	**1st measure [°C]**	**2nd measure [°C]**	**3rd measure [°C]**	**Average [°C]**
1	22,9	39,9	38,9	42,1	40,3
2	22,9	48,8	48,4	48,8	48,7
3	22,9	41,6	41,5	41,5	41,5
4	22,9	42,1	42,6	42,4	42,4

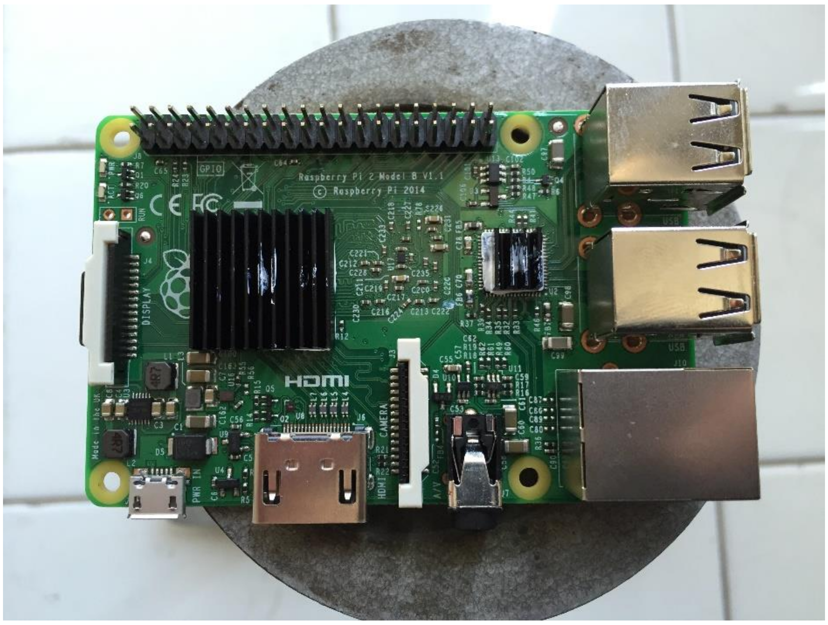

Figure 25. Front face of the Raspberry Pi with the fins mounted.

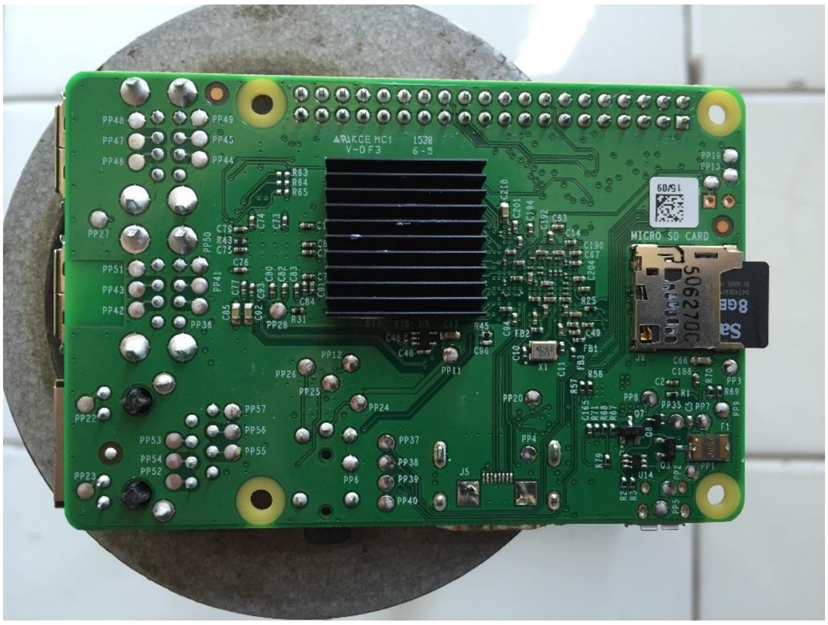

Figure 26. Rear face of the Raspberry Pi with the fins mounted.

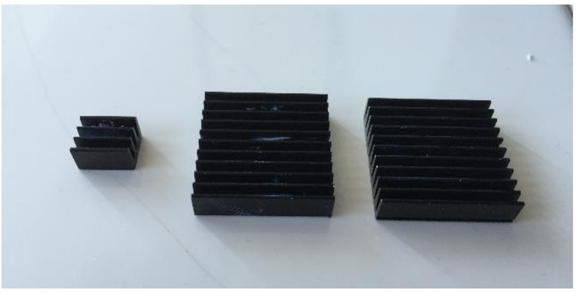

Figure 27. Aluminum fins used in this project.

The dimensions and parameters of the fines are presented in Table 12:

Table 12. Dimensions of the the sets of fins.

	Fins 2 and 4 *	Fins 3*
Length [m^2]	0,019	0,008
Width [m]	0,019	0,007
Height [m]	0,005	0,005
Nº of fins	10,000	4,000
Hipotenuse [m]	0,005	0,005
Width of 1 fin [m]	0,002	0,002
Surface 1 fin [m^2]	1,93E-10	8,10E-11
Surface set [m^2]	0,001934	0,000324

* Means that two sets of the same dimensions were located in the zones 2 and 4 and a smaller one in the zone 3.

To measure the surface of one fin it has been used the equation (41):

$$A_f = 2 \cdot w \cdot \left(L^2 + \frac{\delta^2}{2}\right)^{0.5} \quad (41)$$

Where,

- w : Length of the fin $[m]$.
- L : Height $[m]$.
- $\dot{\delta}$: Width of the fin $[m]$

The previous results allowed us to determine which components are the ones that heat up the most. Those components will be the ones that will require a special attention when designing the cooling system. The components that showed the highest temperature in order were;

1. Processor
2. USB hub
3. RAM
4. Bakelite

It was predictable that the processor became the one with the highest temperature. After all, it is the one that processes the major amount of data. The following Table 13 summarizes the efficiency of the fins in terms of temperature dissipation.

Table 13. Percentage of temperature drop when using fins.

Component	Without fins [°C]	With fins [°C]	Drop of the temperature
Front face			
1	41,7	40,7	2,46%
2	55,4	52	6,54%
3	45,8	45,3	1,10%
4	42,4	41,3	2,66%
Rear face			
1	41,1	40,3	1,99%
2	51	48,7	4,72%
3	43,8	41,5	5,54%
4	45,1	42,4	6,37%

It is clear from the table that the fact of using fins helps in terms of cooling. The drop on the temperature is not really high in this case (the maxim decrease was on the processor in the front face 6,54%) but it must be remembered that there was not used any kind of cooling system. This experiment was executed in a free convection state.

9.4 Power generation

The thermal generation of the Raspberry Pi can be easily calculated using the Newton's law of cooling and adding the radiation as shows the power balance in equation (42):

$$q_{generate} = q_{convection} + q_{radiation} = \sum_{i=1}^{3} q_i + q_{bakelite} \tag{42}$$

Where i is the index for each one of the three components studied.

It has been calculated the power from each surface that has been measured its average temperature (Table 8 and Table 9). This thermal power must be understood as the sum of the power of the front and rear face. Using the equation (5) and equation (6) the heat transfer coefficient in natural conditions in front and rear face respectively can be obtained. Once the coefficient is known the power generation per area unit can be determined considering the previous power balance and using equation (1) and equation (21) for the convective power and radiative power respectively (Table 14 and Table 15).

Table 14. Average temperatures and heat convection coeficient.

Zone	T_{surf_up} [°C]	T_{surf_down}[°C]	T_{bulk} [°C]	L [m]	h_{c_up} [W/m^{2}K]	h_{c_down} [W/m^{2}K]
1	41,733	41,067	23,200	0,017	7,569	3,352
2	55,400	50,967	23,200	0,014	9,141	3,937
3	45,800	43,800	23,200	0,009	9,344	4,081
4	42,367	45,100	23,200	0,012	8,345	3,856

Table 15. Thermal power per area unit.

Zone	q_{front_face} [W/m^2]	q_{rear_face} [W/m^2]	q_{rad_up} [W/m^2]	q_{rad_down} [W/m^2]	q_{total} [W/m^2]
1	140,280	59,894	112,289	107,890	420,353
2	294,348	109,326	223,353	188,409	815,437
3	211,178	84,067	149,454	134,874	579,573
4	159,942	84,452	124,594	144,320	513,308

The results of this table corroborate the thermal image of the Raspberry (Figure 22). The microprocessor is the one that generates more power and consequently is the one that evacuates more per area unit through the convection and radiation mechanism as Table 16 shows.

Table 16. Raspberry PI2 power generation

Zone		Area** [m^2]	q_{total} [W/m^2]	$q_{\_Raspb}$ [W]
1	**Bakelite**	0,00451	420,353	1,894
2	**Processor**	0,00020	815,437	0,160
3	**USB hub**	0,00008	579,573	0,047
4	**RAM**	0,00014	513,308	0,074
Total		**0,005**	**2328,671**	**2,175**

The power calculation has been performed multiplying the area of each zone by the power per unit area of the zone.

Once again the results are coherent with the measurements in the previous section. The power per area unit agree with the temperatures. The component that generate more power is the processor.

It has also been performed the power calculation of the Raspberry Pi when the three sets of fins were mounted. The results are available on Table 17, Table 18 and Table 19

Table 17. Average temperatures using fins and heat convection coeficient.

Zone	T_{surf_up} [°C]	T_{surf_down}[°C]	T_{bulk} [°C]	L [m]	hc_{up} [W/(m²K)]	hc_{down} [W/(m²K)]
1	37,333	39,533	24,700	0,087	4,580	2,131
2	49,967	47,667	24,700	0,014	8,604	3,755
3	41,333	40,533	24,700	0,009	8,655	3,821
4	40,033	41,367	24,700	0,012	7,892	3,602

Table 18. Thermal power per area unit using fins.

Zone	q_{front_face} [W/m²]	q_{rear_face} [W/m²]	q_{rad_up} [W/m²]	q_{rad_down} [W/m²]
1	57,855	31,606	75,421	89,534
2	217,384	86,237	171,800	154,390
3	143,959	60,500	108,347	102,725
4	121,011	60,030	99,233	108,582

Table 19. Thermal power of the front and rear face and total.

Zone	Area rear face [m²]	q_{rear_face} [W]	Area front face [m²]	q_{front_face} [W]	q_{total} [W]
1	0,00451	0,546	0,00451	0,601	1,146
2	0,00020	0,047	0,00193	0,730	0,777
3	0,00032	0,052	0,00008	0,020	0,073
4	0,00014	0,024	0,00193	0,422	0,446
Total	0,00517	0,670	0,008	1,773	**2,442**

In this case it has been compulsory to study the front and rear face power separately. This is because it is not correct to consider the same surface from the rear and front face. The use of fins has increased the area of those zones where they were mounted (zones 2, 3 and 4).

It has to be mentioned that, in order to simplify calculations, the radiative heat flux has been evaluated following equation (21). It corresponds to the radiative heat flux between a planar surface and its surroundings. Thus, small discrepancies in the total generated heat with and without fins are reasonable.

9.5 Experimental h_c calculations

This section includes the experimental calculation of the thermal convection coefficient with and without fins in forced convection and its comparison with the theoretic coefficient from the literature. In order to calculate this coefficient, a small wind tunnel already available at the university MiMT laboratory was used. In the tunnel, both the heat flux and the wind velocity, could be regulated.

This wind tunnel consists in an open conduct in which the air flows through it in different velocities (Figure 28). There is also installed a surface which is heated using an electric resistance with a potentiometer.

Figure 28. Wind pipe device.

From the previous experiment we obtained the power generation of the Raspberry Pi per area unit necessary for this one in order to calculate the heat convection coefficient in forced circulation.

The heated plate was a board whose dimensions were 10x11cm. Multiplying the power per area unit obtained of each of the components by the surface of the plate we were able to calculate the power that we had to introduce in the device every time (Table 20).

Table 20. Power introduced on the deivce.

Zone	Area* [m^2]	$q_{\_resistance}$ [W]
1	0,011	4,624
2	0,011	8,970
3	0,011	6,375
4	0,011	5,646
Total		**25,615**

The calculation of the heat transfer convection in forced circulation has also been performed using the set of fins in order to evaluate the decrease on the surface temperature. In the next section it will be explained how they were located in the plate and why. The power, nonetheless, introduced on the potentiometer in order to perform the h_c calculations was the one from Table 20.

Due to the places were the fins were located were zones 2, 3 and 4 it has only been studied the h_c in those zones. It has been determined that calculate the h_c in zone 1 (bakelite) when using fins was of no interest because in first place it was the zone where the temperatures were lower in the free convection experiments and second the main objective is to study the difference in those zones where the fins were mounted.

9.5.1 Experimental h_c without fins

The experimental heat convection coefficient is determined using the equation (1). In order to obtain the temperatures required in the equation it was used the NTC altogether with the Arduino as a data logger to determinate the air temperature and a Type K thermocouple to measure the surface temperature (Figure 29). Due to the fan could only proportionate a top speed of the fluid of 2m/s it has been analyzed the behavior of the heat transfer coefficient with three different velocities: 1m/s, 1,5m/s and 2m/s. The results of the calculations are gathered on Table 21, Table 22 and Table 23. In order to obtain the results from the steady state the potentiometer was turned on the night before the experiment. By doing this we ensured the validity of the results.

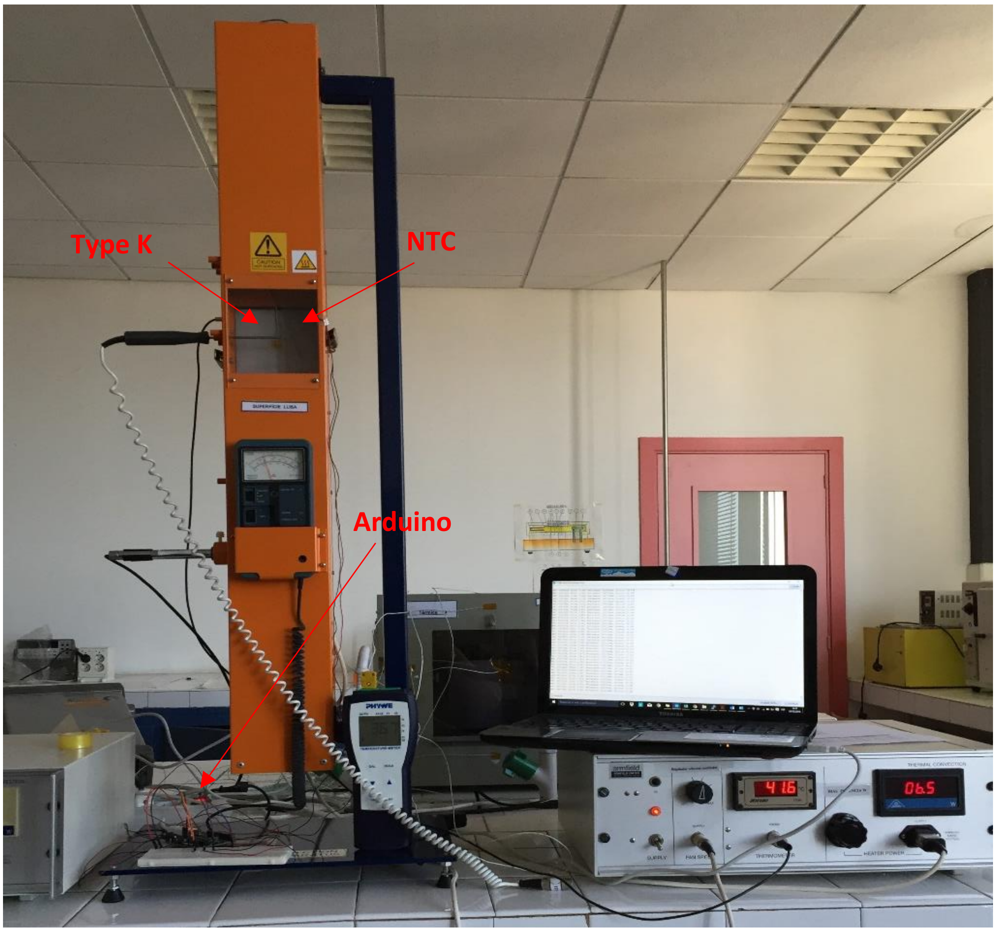

Figure 29. Experimental device with the NTC and the Type K mounted (the type K it's in the middle of the plate, it has been considered that the whole surface has the same temperature. The NTC is located in the middle between the plate an the glass square).

Table 21. Experimental h_c at fluid velocity 1m/s.

v= 1m/s				
Zones	T_{bulk} [C°]	$T_{surface}$ [C°]	q/A [W/(m²K)]	h_c [W/(m²K)]
1	23,6	36,3	420,4	33,1
2	25,1	50,7	815,4	31,9
3	23,9	42,2	579,6	31,7
4	23,9	40,1	513,3	31,7

Table 22. Experimental h_c at fluid velocity 1,5m/s.

v= 1,5m/s				
Zones	T_{bulk} [C°]	$T_{surface}$ [C°]	q/A [W/(m²K)]	h_c [W/(m²K)]
1	24,5	35,6	420,4	37,9
2	24,2	46,6	815,4	36,4
3	24,1	39,9	579,6	36,7
4	23,7	37,2	513,3	38,0

ETSEIB

Table 23. Experimental h_c at fluid velocity 2m/s.

v= 2m/s				
Zones	T_{bulk} [C°]	$T_{surface}$ [C°]	q/A [W/(m²K)]	h_c [W/(m²K)]
1	24,5	34	420,4	44,3
2	24,2	44,2	815,4	40,77
3	23,8	37,4	579,6	42,61
4	24	35,7	513,3	43,9

The temperatures when running the experiment in forced convection are lower than the ones in free convection. This is of particularly interest for our project because it will determine the viability of it. The following Table 24 summarizes the drop of temperature using the average temperature of the front and rear face in the free convection experiment (Table 8 and Table 9).

Table 24. Percentage of temperature drop in forced convection.

	Free convection [°C]	Forced convection [°C]					
		v=1 m/s		v=1,5 m/s		v=2 m/s	
Bakelite	41,4	36,3	12,32%	35,6	14,01%	34	17,87%
Processor	53,2	50,7	4,70%	46,6	12,41%	44,2	16,92%
USB hub	44,8	42,2	5,80%	39,9	10,94%	37,4	16,52%
RAM	43,75	40,1	8,34%	37,2	14,97%	35,7	18,40%

The table shows a max decrease of the temperature of almost a 20% when the fluid velocity is 2 m/s. This is crucial when designing the cooling system because in order to make the cluster the most efficient possible it must be considered the electric consumption. Higher fluid velocity means more consumption.

However it is worth to mention the uncertainty factor of the instruments. The uncertainty calculated for the NTC used in the experiment was 0,89 °C and for the type K is, as shown in Annex C.8, 0,5°C. Apart from the uncertainty there are other factors that might have distorted the temperature taken from the visualizer such as the power supplied from the potentiometer. Although it was assumed that after 30 minutes of the change of the air velocity the system would be stable the visualizer of the potentiometer was oscillating 0,2W up 0,2W down. In the worst case, however, the results of the heat transfer coefficient would only vary 3-6% its value.

9.5.2 Experimental h_c with fins

The h_c of the set of fins has been calculated taking into account the following conditions:

- It has only been performed the calculations of the zones where the fins were mounted (zones 2, 3 and 4).
- The fins were placed in the middle of the heated plate because in the inside of it there was a type K thermocouple whose measure could be used in further calculations.

- The power introduced on the potentiometer was the one used in the without fins experiment. It was done this way because we wanted to study the difference in terms of thermal power dissipation when using fins so it had to be used the same power of the without fins experiment. Then, in the calculation of the experimental h_c with fins using the equation (1), the q had to be calculated considerating the surface of the set of fins.

A good way to know which power was dissipated by the set of fins is using an energy balance in first place:

$$q_{generated\_1\rightarrow 2} = q_{conduction\_2\rightarrow 3}$$

The next step is to consider the diffusion law of a plate surface with generation. Figure 30 shows the disposition of the sets and the temperatures needed.

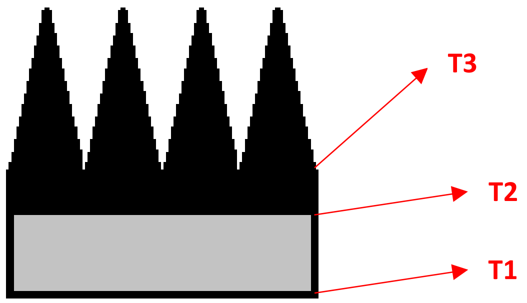

Figure 30. The Black zone represent the set of fins and the grey zone represent the thickness of the heated plate.

T3 was the one provided with the type K thermocouple and the T1 was provided by the device type K thermocouple that has in his inside. So the unique unknown is T2. The diffusion law for plate surface presented in equation (43) can provide us with that temperature.

$$0 = \lambda_{1\rightarrow 2} \cdot \frac{d^2T}{dx^2} + \dot{g} \tag{43}$$

Where,

- $\lambda_{1\rightarrow 2}$: The conductivity of the steel $[W \cdot m^{-1} \cdot K^{-1}]$.
- $\frac{d^2T}{dx^2}$: Temperature gradient.
- $\dot{g}$: Generation $[W \cdot m^{-3}]$

And now integrating two times and imposing two boundary conditions we obtain equation (44) which serves to calculate the temperature needed:

$$T(x) = -\frac{\dot{g} \cdot x^2}{\lambda_{1\rightarrow 2} \cdot 2} + C_1 \cdot x + C_2 \tag{44}$$

$$c.c\ 1 \ \rightarrow \ \frac{dT}{dx} = 0 \quad when \quad x = 0$$

$$c.c\ 2 \ \rightarrow \ T(x) = T1 \quad when \quad x = 0$$

Unfortunately the precision and stability of the instruments available were not high enough and the results of T1 and T3 were in some occasions contradictory (the temperature T3 was higher than T1).

So we had to search for a simpler method. The objective of the previous one was to determine the real power that flows through the fin. And that power is mainly regulated by the surface of the fin. Inferring a simple rule of three with the surface and the power the results were less accurate but coherent, the one that dissipated more was still the processor.

The experimental conditions were exactly the same ones from the without fins experiment. The potentiometer was turned on the night before in order to achieve the steady state of the power and then the fluid velocity was fixed. In intervals of 30 minutes the temperature was taken and the velocity was changed. The results of this experiments are shown in Table 25, Table 26 and Table 27 :

Table 25. Experimental h_c at fluid velocity 1m/s using fins.

v= 1m/s						
Zones	T_{bulk} [K]	$T_{surface}$ [K]	q [W]	A [m^2]	ef	h_c [W/(m^2k)]
2 = Processor	25,3	48,1	1,577	0,002	0,998	35,82
3 = USB hub	25,1	40	0,188	0,0003	0,998	39,00
4 = RAM	24,5	38,4	0,993	0,002	0,999	36,99

Table 26. Experimental h_c at fluid velocity 1,5m/s using fins.

v= 1,5 m/s						
Zones	T_{bulk} [K]	$T_{surface}$ [K]	q [W]	A [m^2]	ef	h_c [W/(m^2k)]
2	23,6	43,5	1,577	0,002	0,998	41,05
3	25,3	37,3	0,188	0,0003	0,997	48,45
4	24,5	35,7	0,993	0,002	0,998	45,91

Table 27. Experimental h_c at fluid velocity 2m/s using fins.

v= 2 m/s						
Zones	T_{bulk} [K]	$T_{surface}$ [K]	q [W]	A [m^2]	ef	h_c [W/(m^2k)]
2	22,7	42,3	1,577	0,002	0,998	41,67
3	24,2	36,8	0,188	0,0003	0,997	46,14
4	23,9	34,3	0,993	0,002	0,998	49,45

The following Table 28 was used when calculating the h_c due to the calculation of the m parameter required for the h_c value. In order to use it properly what has been done is an iteration of the results until achieve a good approximation. The iterations stopped when the differences were less than 0,001 units.

Table 28. Fins parameters needed to perform the h_c calculation.

	v = 1m/s		v = 1,5 m/s		v = 2 m/s		v =1 m/s	v=1,5 m/s	v = 2 m/s
	Zone 2	**Zone 4**	**Zone 2**	**Zone 4**	**Zone 2**	**Zone 4**	**Zone 3**	**Zone 3**	**Zone 3**
P [m]	0,0584	0,0584	0,0584	0,0584	0,0584	0,0584	0,0363	0,0363	0,0363
A [m²]	9,67E-05	9,67E-05	9,67E-05	9,67E-05	9,67E-05	9,67E-05	4,05E-05	4,05E-05	4,05E-05
h [W/(m²k)]	41,849	37,587	51,470	41,849	41,849	41,849	39,210	48,711	46,385
λ [W/(m·K)]	205	205	205	205	205	205	205	205	205
m	11,100	10,519	12,309	11,100	11,100	11,100	13,082	14,582	14,229
ef	0,998	0,999	0,998	0,998	0,998	0,998	0,998	0,997	0,997

9.5.3 Theoretic h_c

The theoretic value for the heat transfer coefficient can be obtained from several empirical correlations. Those from pipelines explained in section 6.1.4 were discarded due to the poor accuracy of the problem geometry. The fact that both of them were from pipelines in where the hot surface is the whole pipe not just a small section that does not even surround the whole perimeter lead us to refuse both of them. However, in order to verify this presupposition the calculation of the heat convection coefficient using the Gnielinsky correlation can be found on Annex D.4. Bittus-Boelter correlation was not performed due to the Reynolds number doesn't meet the conditions of use.

Despite that pipeline correlations are not viable for our problem, another kind had to be used. Due to it was used a flat plate, a very good correlation is the one that can be found in *Incropera, Chapter 7, Section 7.2.5 Flat plates with constant heat flux conditions* [12]. This correlation can be used when there is an unheated starting length. This can be assimilated to our case due to the hot plate is not at the beginning of the tube but at the half of it. This correlation is used for plates and is acceptable to be used in our circumstance: when the power is constant.

$$Nu = 0.68 \cdot Re^{0.5} \cdot Pr^{\frac{1}{3}} \tag{45}$$

Then using equation (1) the h_c can be obtained. Two very important remarks for this correlation are:

- The length used in order to calculate the Re has been determined using the middle point of the hot plate starting from the bottom of the pipeline due to the hydrodynamic boundary layer starts at the very beginning of it. This was 0,60m.
- The length used when calculating the Nusselt is 0,1 and in this case is the length of the plate.

The whole data needed for this correlation is available in Annex D.3. The main results of it are presented in Table 29, Table 30 and Table 31:

Table 29. h_c calculation when velocity 1m/s.

Zone	T_{bulk} [°C]	$T_{surface}$ [°C]	Re	Nu_L	Nu	h_c [W/(m^2·K)]
1	23,6	36,3	3,88E+04	0,10	119,869	30,89
2	25,1	50,7	3,84E+04	0,10	119,309	30,87
3	23,9	42,2	3,87E+04	0,10	119,757	30,88
4	23,9	40,1	3,87E+04	0,10	119,757	30,88

Table 30. h_c calculation when velocity 1,5 m/s.

Zone	T_{bulk} [°C]	$T_{surface}$ [°C]	Re	Nu_L	Nu	h_c [W/(m^2·K)]
1	24,5	35,6	5,79E+04	0,10	146,397	37,82
2	24,2	46,6	5,80E+04	0,10	146,534	37,82
3	24,1	39,9	5,80E+04	0,10	146,580	37,82
4	23,7	37,2	5,82E+04	0,10	146,763	37,83

Table 31. h_c calculation when velocity 2 m/s.

Zone	T_{bulk} [°C]	$T_{surface}$ [°C]	Re	Nu_L	Nu	h_c [W/(m^2·K)]
1	24,5	34	7,72E+04	0,10	169,045	43,67
2	24,2	44,2	7,73E+04	0,10	169,203	43,67
3	23,8	37,4	7,75E+04	0,10	169,414	43,68
4	24	35,7	7,74E+04	0,10	169,309	43,68

The results obtained using the correlations are really similar to the ones obtained experimentally. The following Table 32 informs of how much different the theoretic ones are from the experimental ones. As we can see, the maximum difference is when the velocity was fixed at 2 m/s with an error of 7,11%. It has been considered that this value is not big enough to invalidate the theoretic results. It is usually an error on the heat transfer coefficient of 20% when the results must be discarded.

Table 32. Comparative table between the experimental heat transfer coefficient and the theoretic one.

Experimental h_c [$W/(m^2{\cdot}K)$]	Theoretic h_c [$W/(m^2{\cdot}K)$]	Error
v=1 m/s		
33,1	30,89	6,68%
31,9	30,87	3,23%
31,7	30,88	2,59%
31,7	30,88	2,59%
v=1,5 m/s		
37,9	37,82	0,21%
36,4	37,82	3,90%
36,7	37,82	3,05%
38	37,83	0,45%
v=2 m/s		
44,3	43,67	1,42%
40,77	43,67	7,11%
42,61	43,68	2,51%
43,9	43,68	0,50%

10 Study of the sustainability of the project

Based on the guidelines set out in Law 21/2013, of December 9, Environmental Assessment [27] it has been carried out the study of sustainability of this project.

10.1 General description of the project

This project consists, in the experimental characterization of a Raspberry Pi. So basically the study of sustainability takes into account carrying out this task.

10.2 Description of the environment

This project was conducted entirely in MiMT laboratory, located in ETSEIB. During this period of time some computers of this laboratory and some devices and instruments of it have been used. This has created a social impact as the laboratory staff couldn't count on all the materials while the experiments of this project were being carried out.

10.3 Assessment of impacts and identification

This section explores the various potential impacts that may occur on the environment during the project. The cause of these impacts may be the existence of the project, the use of natural resources and the emission of pollutants.

Moreover, when the study of sustainability of a project, it is divided into a main phases to consider: construction, production and decommissioning activities.

10.3.1 Criteria

Below are a series of definitions and techniques for assessing the environmental impact. These terms refer to different types of effects arising from the impact of a project and make different classifications depending on the criteria. Therefore:

Depending on the type of effect:

- ✓ *Positive effect*: The one that is admitted by both the scientific community and the general population in the context of a comprehensive analysis of the costs and benefits of generic contingencies external action contemplated.
- ✓ *Negative effect*: That which results in the loss of natural value, aesthetic and cultural landscape, ecological productivity; or an increase in pollution damage arising from erosion and other environmental hazards in disagreement with the eco-geographical structure, character and personality of a given location.

Depending on the impact of the effect:

- ✓ *Direct effect*: The one that has an immediate impact on some aspect of the environment.
- ✓ *Indirect or secondary effect*: The one that suppose an immediate impact with regard to interdependence or, in general, with respect to a relationship with another environmental sector.

The following section contains various definitions of the magnitude of the assessment of a potential environmental impact:

- *Compatible IA:* one in which the recovery is immediate at the end of the activity and does not require protective measures.
- *Moderate IA:* one in which the recovery does not require intensive corrective or protective measures, but to recover the initial conditions require a certain time.
- *IA Sever*: one in which the recovery of environmental conditions requires the adequacy of corrective or protective measures, plus an extended period of time.
- *Critical IA:* those with a magnitude greater than the acceptable limit, resulting in a permanent loss of the quality of environmental conditions, without the possibility of recovery, even with the protective or corrective measures.

In addition to assessing the impacts of different specific cause-effect relationships, they must be evaluated with respect to the overall impact of the project. The overall magnitude of the project will be positive if the overall assessment is supported, moderate or severe, while it will be negative if the overall assessment is critical.

10.3.2 Potential environmental impacts

The cause of the environmental impacts, as has been mentioned above, can be due to the existence of the project, the use of the natural resources or the emission of pollutants.

Therefore, it has been performed a study to analyse the impact caused by the different reasons.

Impact due to the existence of the project

The realization of the project turns into a positive effect. Some experiments have been carried out in order to obtain the desired results. However in those experiments the environment has been harmed the less as possible. Basically it has been consumed electricity and thus it has been produced thermal generation.

On the other hand, the study of this project contributes into the development of a more sustainable super computer in terms of efficiency and energy consumption. The study will help the developers to design a better cooling systems which will be traduced in a better performance of the computers and into a less heat production of the internal fans at the same time.

Impact due to the use of resources

A direct impact is caused by the use of resources. This resources are the devices used for the experiments and also the office items (scissors, papers, ruler, ballpoint pens, adhesive tape…). All of them during its manufacture contribute to the environmental degradation in more or less extent. However most of those materials can be recycled. Waste management is carried out as set in Table 33 which also identifies all waste generated according to the rules required by the Catalan Waste [28].

Table 33. Classification of the generated waste

Code	Waste	Process in which is generated	Management
200101	Paper	Printing and hand annotations	Recycling container blue
080309	Printing ink	Printing	Container toners and printer cartridges that are in each school building.
200199	Office items	During the elaboration of the project	Recycling container yellow
160214	RAEE	During the elaboration of the project	Selective management system of the manufacturer

Impact due to the emissions

It is considered as an indirect impact derived from the consumption of electric energy that supposes an emission of CO_2 gases to the atmosphere in the thermic centrals. The Table 34 performs an average calculation of those gases based in the electric consumption.

Table 34. Impact due to the emissions calculations

	Power consumption [W]	**Average time of usage [hours]**	**Power consumption [kWh]**	**Emission of CO_2 Kg per kWh produced [$KgCO_2$/kWh]** [29]	**Kg of CO_2****
Computer with a I5-2450M processor	35[30]	315	11,025	0,33	3,64
Raspberry Pi	2 [31]	6*	0,012	0,33	3,96e-3
Total					3,64

**It has been calculated taking into account the performances done in order to detect the hot spots. This test has been done 4 times and it took 1,5 hours each. A total of 6 hours.*

*** It has been considered that the energy produced is equivalent to the energy consumed without any kind of losses during the transport.*

There has been a generation of 3,64Kg of CO_2 during the development of this project.

All the impacts of this study can be qualified as compatible and, therefore, the overall impact associated with the project as well.

11 Financial analysis

This section shows the costs incurred during the project, which are divided into two categories, the costs of the material resources and the costs of the human resources.

11.1 Material resources

The materials used through this project has been listed in Table 35 and Table 36. Some of them have been bought specifically for this project. However some other were already available on the laboratory. For those ones it has been considered a useful life of 10 years and has been taken the proportional cost for 5 months of use.

Table 35. Cost of the material bought for this project

Material	Cost [€]
Arduini Uno	19,82
Raspberry PI 2	41,26
Rohs BC thermistor NTC 10kOhm	2,15
Raspberry PI 2 toolkit	10,41
10 kohm Resistor(50 units)	1
TOTAL	**74,64**

Table 36. Cost of the amortization of the laboratory devices.

Material	Shop price [€]	Price/month [€]	Project cost [€]
Sensor type K	82,65	0,69	3,44
Chronometer	3,64	0,03	0,15
Oven HK50	1190,40	9,92	49,60
Thermocouple visualizer	523,00	4,36	21,79
TOTAL			**74,99**

TOTAL cost of material resources [€]	**149,63**

11.2 Human resources

In this section we include the hours of the person in charge of the project and the three other people who have taken an active performing management tasks.

It is considered a cost € 20 / hour for the junior engineer who made the project, which is devoted an average of 15 hours per week for 21 weeks. It involves both the experimental and theory tasks. The direction of the project was given by two research engineers, with a commitment of 2 hours per week. The fees estimated by the research staff is 30 € / hour. During the experimental part a researcher in training was involved in the project. The average time dedicated of this was 1,5 hours per week with an estimated fee of 25 €/hour.

With the information above, the cost of this project derived from human resources is shown in Table 37.

Table 37. Human resources costs.

Staff involved	**Cost per hour [€/hour]**	**Hours worked**	**Total cost [€]**
Junior engineer	20	315	6.300
Research engineer (director)	30	41	1.230
Research engineer (codirector)	30	41	1.230
Associate researcher	25	31,5	787,5
TOTAL			**9.547,5**

11.3 Total cost

The following Table 38 shows the total cost of the project.

Table 38. Total cost of the project.

Concept	**Cost [€]**
Material resources	149,63
Human resources	9.547,5
SUBTOTAL	9.697,13
Unforeseen expenses (10%)	969,71
TOTAL	**10.666,84**

12 Results and conclusions

In the frame of present project, the Raspberry Pi 2 has been successfully characterized thermally and the feasibility of using available correlations with fins as cooling technology has been proven. The main conclusions and goals achieved can be listed as following:

1. The methodology to measure temperatures with NTC thermistor and using an Arduino as a data logger has been learned and successfully tested
2. It is really important whilst running experiments on the laboratory to have everything stabilized in order to obtain accurate results. This was particularly important in our case due to the long waits in some of the experiments realized. A slightly modification in the air temperature could change the results of the whole experiment. The equipment and devices used to measure the magnitudes needed for this project sometimes tend to fluctuate. Hence, it is really important to pay special attention in the timing and the conditions that each of the experiments require.
3. The NTC thermistor has been calibrated and the used methodology can be easily reproduced for other NTC devices. However some aspects should be considered such as the need of a very well insulating system in order to only measure the temperature of interest.
4. NTC thermistor has been proven to be useful when it is measuring in the range of temperatures where the resistance has to vary a lot in order to increase/decrease the temperature a few degrees. This zone is always on the left side of the characteristic curve of the NTC.
5. Hot spots have been identified, being the critical one the processor. It is coherent that the processor is the one that heats up the most because in fact is where all the data is processed and calculations are performed.
6. The total generated heat along the Raspberry Pi has been calculated.
7. The use of fins was not only positive in terms of cooling down the temperature of the main chips of the device but also the whole temperature of the chip. The power generated by the Raspberry is the same regardless of using the fins or not so if the power evacuated by the set of fins was higher that means that the rest of the surface of the device had to evacuate less.
8. The heat transfer coefficient with fins in natural convection is similar to the without fins one. Same happens when there is forced convection. Considering equation (1) the following conclusion can be inferred: the fact of using fins increases the difference of temperature between the bulk and the surface, however this difference is compensated by the increase of area due to the use of fins. So all in all, the heat transfer coefficient is compensated and remains more less the same.
9. With the obtained know how and information of this project it is convenient to keep tasting the use of fins in different conditions (p.e. different flow velocities, different air temperatures). It will also be suitable to try to run those experiments with Raspberry's mounted in parallel in order to start working with the first prototype of cluster.

13 Acknowledgments

I would like to thank my director, Vicente, and co-director, Elisabet, for their assistance and patience throughout the course of this project providing me with data and valuable information about how I should consider certain sections. Especially during the last simulations of the project that were the ones in which we had more work to do.

I thank Nuria for her willingness of help when I needed to use the Laboratory or any of the devices/materials required in order to proceed with the experiments.

To my family and friends for helping me to succeed in this bachelor thesis.

14 Bibliography

[1] “Raspberry Pi Stress Tests and Burn-In Apps - Roy Longbottom’s Benchmark Collection.” [Online]. Available: http://www.roylongbottom.org.uk/Raspberry Pi Stress Tests.htm#anchor7. [Accessed: 01-May-2016].

[2] “What is Computer Cluster? - Definition from Techopedia.” [Online]. Available: https://www.techopedia.com/definition/6581/computer-cluster. [Accessed: 21-Apr-2016].

[3] G. Pfister, *In Search of Clusters*. Upper Saddle River, NJ: Prentice Hall PTR, 1998.

[4] “OLD-COMPUTERS.COM : HISTORY / detailed info.” [Online]. Available: http://www.old-computers.com/history/detail.asp?n=20&t=3. [Accessed: 21-Apr-2016].

[5] James Pearn and J. Pearn, *How many servers does Google have?* plus.google.com.

[6] “Convective Heat Transfer Convection Equation and Calculator | Engineers Edge | www.engineersedge.com.” [Online]. Available: http://www.engineersedge.com/heat_transfer/convection.htm. [Accessed: 04-May-2016].

[7] “The Nusselt Number.” [Online]. Available: http://pages.jh.edu/~virtlab/heat/nusselt/nusselt.htm. [Accessed: 24-Apr-2016].

[8] H. E. Zellnik and S. W. Churchill, “Convective Heat Transfer from High- temperature Air Inside a Tube,” vol. 4, no. 1.

[9] D. Base, “Grau Enginyeria en Tecnologies Industrials (240171),” no. 240171, pp. 1–16.

[10] R. S. Subramanian, “Heat transfer in Flow Through Conduits,” *Convect. tubes*, vol. 1, pp. 1–9, 2006.

[11] O. Levenspiel, *Engineering Flow and Heat Exchange, Revised Edition*, no. 2. Plenum Press, 1998.

[12] D. P. . Incropera, Frank; Dewitt, *Heat and Mass Transfer*. 2011.

[13] “Arduino - Introduction.” [Online]. Available: https://www.arduino.cc/en/Guide/Introduction. [Accessed: 19-Mar-2016].

[14] “Arduino - Environment.” [Online]. Available: https://www.arduino.cc/en/Guide/Environment. [Accessed: 12-Jan-2016].

[15] “What is an Arduino? - learn.sparkfun.com.” [Online]. Available: https://learn.sparkfun.com/tutorials/what-is-an-arduino. [Accessed: 12-Jan-2016].

[16] “Microcontroller Comparison.” [Online]. Available: http://www.makershed.com/pages/microcontroller-comparison. [Accessed: 06-Feb-2016].

[17] “Measuring the Temperature With an Arduino and a Thermistor - iwantmyreal.name.” [Online]. Available: http://iwantmyreal.name/blog/2012/09/23/measuring-the-temperature-with-an-arduino-and-a-thermistor/. [Accessed: 12-Apr-2016].

[18] “Using a Thermistor | Thermistor | Adafruit Learning System.” [Online]. Available: https://learn.adafruit.com/thermistor/using-a-thermistor. [Accessed: 12-Apr-2016].

[19] Gregg Lavenuta, “AN EXPLANATION OF THE BETA AND STEINHART-HART EQUATIONS FOR REPRESENTING THE RESISTANCE VS . TEMPERATURE RELATIONSHIP IN NTC THERMISTOR MATERIALS Gregg Lavenuta , QTI Sensing Solutions Engineering Department,” *QTI Sens. Solut. Eng. Dep.*

[20] A. Note, “Comments on the Steinhart-Hart Equation Comments on the Steinhart-Hart Equation,” pp. 2–5, 2015.

[21] I. Spectrum Sensors & Controls, “NTC Thermistor Engineering Notes,” pp. 49–52, 2005.

[22] S. Bell, “A Beginner’s Guide to Uncertainty of Measurement,” *Meas. Good Pract. Guid.*, no. 2, p. 41, 1999.

[23] E. L. Equipo, D. E. G. D. E. La, C. Del, C. Nacional, D. E. S. Publica, and P. Las, “Cálculo de la incertidumbre de la medición,” 2008.

[24] “Raspberry Pi Blog - News, Announcements, and Ideas.” [Online]. Available: https://www.raspberrypi.org/blog/. [Accessed: 21-Apr-2016].

[25] “Raspberry Pi 2 Model B | Raspberry Pi 2 Model B | Raspberry Pi.” [Online]. Available: http://es.rs-online.com/web/p/kits-de-desarrollo-de-procesador-y-microcontrolador/8326274/. [Accessed: 21-Apr-2016].

[26] “Raspberry Pi Thermal Photography - Album on Imgur.” [Online]. Available: http://imgur.com/gallery/tzgPU. [Accessed: 04-May-2016].

[27] “Ministerio de Agricultura Alimentación y Medio Ambiente, «Ley 21/2013, de 9 de diciembre, de evaluación ambiental», p. 98151-98227, 2013.”

[28] “Tipus de residu. Agència de Residus de Catalunya. Generalitat de Catalunya.” [Online]. Available: http://residus.gencat.cat/ca/ambits_dactuacio/tipus_de_residu/. [Accessed: 25-May-2016].

[29] Idae, “Propuesta de documento reconocido factores de emisión de CO2,” p. 32, 2014.

[30] “Intel® Core™ i5-2450M Processor (3M Cache, up to 3.10 GHz) Especificaciones.” [Online]. Available: http://ark.intel.com/es-es/products/53452/Intel-Core-i5-2450M-Processor-3M-Cache-up-to-3_10-GHz. [Accessed: 25-May-2016].

[31] “Raspberry Pi • View topic - RasPi power usage measurements ALL Models.” [Online]. Available: https://www.raspberrypi.org/forums/viewtopic.php?f=2&t=6050. [Accessed: 25-May-2016].